THE UNIVERSITY OF QUEENSLAND

Bachelor of Engineering Thesis

Optimising Scramjet Combustor Geometry

Student Name: Marguerite TAYLOR

Course Code: MECH4500

Supervisor: Dr Vincent Wheatley

Submission date: 28 October 2016

A thesis submitted in partial fulfilment of the requirements of the
Bachelor of Engineering degree
in Mechanical Engineering

UQ Engineering

Faculty of Engineering, Architecture and Information Technology

Abstract

Air-breathing propulsion used in hypersonic flight offers many advantages over traditional rocket propulsion, with a range of applications including Low Earth and Sun Synchronous Orbit insertion of small payloads. Scramjets promise greater efficiency; however, they only operate successfully over a portion of the required velocity, so would be best utilised in a multistage vehicle. Scramjet combustors are typically designed to have an initial constant area section to initiate combustion by maintaining high temperatures and pressures. This is followed by a linearly diverging area which expands flow to cool it, alleviating dissociation of the product.

This project aims to determine the optimal geometry, minimising combustor length while maximising heat release by supporting rapid combustion initiation upstream and minimising product dissociation downstream. A literature review was conducted on quasi-one-dimensional scramjet combustor analysis and solution methods. The resulting set of stiff Ordered Differential Equations have been developed into a functional computational method which utilises a multistage reaction mechanism and existing thermo-chemistry routines. This model predicts ignition and allows for adaptation of the geometry, initial conditions and mixing curve.

The Mach 12 REST engine was considered for analysis, with a mixing efficiency curve used to describe the mass flow rate differential. The effects of friction on the results were first examined, followed by consideration of the constant area and constantly diverging area cases. The geometry of the combustor was further varied, changing the location at which the divergence began and the amount it diverges, to examine the effect on flow properties. It was found that the combustion efficiency was greatest when the divergence commenced 15 mm further along the combustor and when the ratio of the final to initial area was maximised.

The results display how area variations can affect the flow properties; as such, further work should consider the development of an optimisation routine into which this computational model can be embedded.

Acknowledgements

The work presented in this thesis would not have been possible without the support and involvement of several others.

First and foremost, I would like to thank my supervisor, Dr Vincent Wheatley. Without his dedicated assistance and support this project would not have been completed. His technical advice and involvement in the project was vital to its success and greatly appreciated. The motivation and encouragement he provided, particularly when things were looking bleak, was invaluable.

I would also like to thank other students who have provided coding assistance along the way, namely John Scolaro and Ryan Kidd, who generously shared their knowledge of Linux OS, Python and C++.

Completing this project required more than academic support, and there are countless people to thank for listening to, and tolerating, me over the course of the project. Namely, Edward Greenaway, whose friendship and support over the course of our university studies will always be appreciated.

Most importantly, I would like to extend my gratitude to my family, especially my parents John and Fiona. I cannot begin to express my gratitude for their encouragement, unwavering support and understanding.

Table of Contents

Table of Figures

Table of Tables

1.0 Introduction

Air-breathing propulsion offers many advantages over traditional rocket propulsion when it comes to hypersonic flight, including higher specific impulse, lower thrust-to-weight ratios and the possibility of recoverable space launchers [1] [2]. For hypersonic speeds (greater than Mach 5) a type of ramjet is used where the combustion takes place at supersonic speeds, these are known as scramjets [2]. Scramjet design contains many difficulties, including: mixing and ignition in short residence times; high heat loads and frictional losses; thermal choking; and incomplete combustion [1]. The limits of scramjet technology are still being determined, however it is estimated that the operational range of scramjets is between Mach 4 and Mach 15 [3].

Scramjets promise greater efficiency than rockets for Low Earth Orbit (LEO) and Sun Synchronous Orbit (SSO) insertion of small payloads [3]; however, they only operate successfully over a portion of the required velocity range. A multistage vehicle would utilise this greater efficiency while still accepting the limitations of air-breathing propulsion. One option for such a system is a three-stage rocket-scramjet-rocket vehicle [3].

Scramjets can be divided into three processes for sectional analysis: compression occurring in the inlet and forebody; combustion occurring in the burner; and expansion in the nozzle [1].

In scramjet combustors, as combustion and heat release proceed, the flow temperature increases such that the product, water, begins to dissociate. To alleviate this, practical scramjet combustors have a linearly diverging area that expands the flow to cool it. The initial section of the combustor however, requires a constant area to maintain the high temperatures and pressures required for initiation of the combustion process [2].

The goal of this project was to determine the optimal geometry that minimises combustor length while maximising heat release, by supporting rapid combustion initiation upstream and minimising product dissociation downstream. The aim was to use computational tools along

with existing flow solvers and reaction mechanisms to develop a scramjet combustor model. This was done through a quasi-one-dimensional analysis of a scramjet combustor, using Ordinary Differential Equations (ODEs) to march along the scramjet length. Existing thermo-chemistry routines were then incorporated into this modelling of the combustor flow field. Originally, the aim was to then develop an optimisation routine in which the scramjet combustor model could be embedded, and thus allow determination of the optimal area profile for a given design. Time constraints resulted in the scope being modified, and the development of the optimisation routine being removed from the project.

2.0 Background

2.1 Area Profiles

Scramjet combustors have typically been designed with a constant area or with a constant area section followed by a constantly diverging section. HyShot II, HIFiRE 2 and SCRAMSPACE are examples of constant area combustors, while REST engines are examples of a constant area section followed by a constantly diverging section. The combustor design of the HyShot II scramjet is shown in Figure 1 [4]; this corresponds to the expected flow field properties plotted against length in Figure 2 [4]. Figure 3 shows a constantly diverging area profile combustor [1]; the flow field properties calculated to correspond to this are shown in Figure 4 [1].

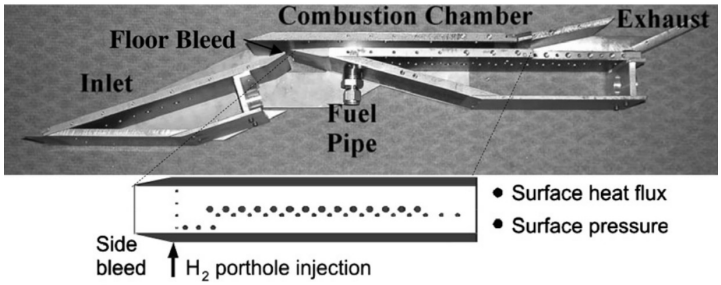

Figure 1: HyShot II Model [4]

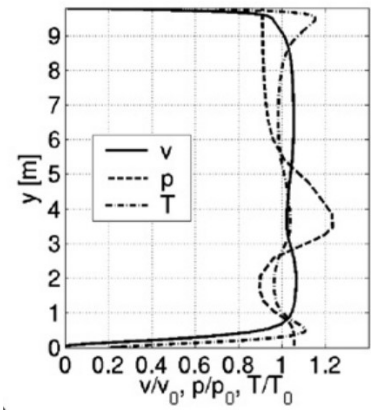

Figure 2: HyShot II flow properties [4]

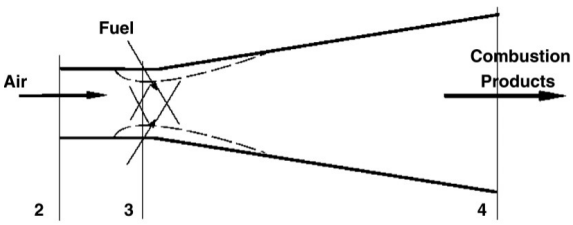

Figure 3: Diverging Area Combustor [1]

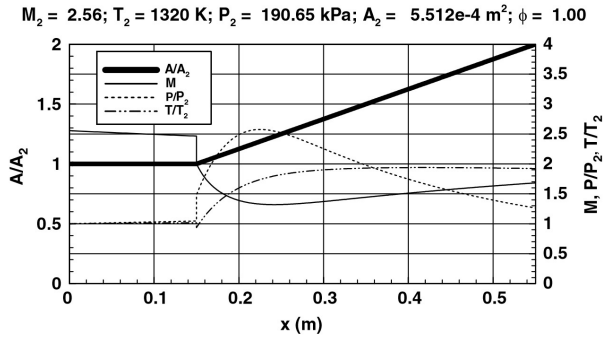

Figure 4: Diverging Area flow properties [1]

Considering the behaviour depicted for these combustor designs, it is clear that the flow field properties are sensitive to changes in area and can be improved upon through determination of an optimal area profile.

2.2 Quasi-One-Dimensional Modelling

Quasi-one-dimensional modelling is a rapid means of predicting the combustor flow field which will make optimisation a less computationally expensive process [5]. Scramjet performance is extremely sensitive to small changes in design, thermodynamic and mixing parameters [2]. Thus, the presented literature focuses on the governing equations of scramjet flow field analysis. A solution methodology presented by [5] and [2] is then reviewed, followed by an analysis of mass mixing profile, specific heats and chemical kinetics.

2.3 Governing Equations

Analysis of a scramjet combustor requires modelling of the processes of fuel addition, fuel and air mixing and combustion [1]. This has previously been done under the assumptions of quasi-one-dimensional, steady-state flow which behaves as a perfect gas [2] [5]. Treating all variables as functions only in the x-direction enables the development of a set of differential equations. A complete list of the variables and nomenclature used is included in Appendix A.

2.3.1 Conservation of Mass

The continuity equation can be expressed in differential form, as seen in Equation 1. It has been observed [2] [5] that this allows for variation in the cross-sectional area profile of the combustor $\left(\frac{dA}{dx}\right)$ and mass injection $\left(\frac{d\dot{m}}{dx}\right)$, which are essential for the project.

$$\frac{1}{\dot{m}}\frac{d\dot{m}}{dx} = \frac{1}{\rho}\frac{d\rho}{dx} + \frac{1}{U}\frac{dU}{dx} + \frac{1}{A}\frac{dA}{dx} \qquad \textit{Equation 1}$$

2.3.2 Conservation of Momentum

The quasi-one dimensional momentum equation in differential form is expressed in Equation 2. The ε accounts for angled fuel injection [5], and is zero for normal injection; [2] omitted this term from its presentation of equations as normal injection was assumed for all cases considered.

$$\frac{1}{p}\frac{dp}{dx}+\frac{\gamma M^2}{2U^2}\frac{dU^2}{dx}+\frac{2\gamma M^2 C_f}{\mathcal{D}}+\frac{\gamma M^2(1-\varepsilon)}{\dot{m}}\frac{d\dot{m}}{dx}=0 \qquad \textit{Equation 2}$$

2.3.3 Equation of State

The equation of state for a perfect gas can also be expressed in differential form, as seen in Equation 3. This allows for the inclusion of chemical changes resulting from mass injection and fuel combustion [5], through use of the mean molecular weight [2], expressed in differential form in Equation 4.

$$\frac{1}{p}\frac{dp}{dx}=\frac{1}{\rho}\frac{d\rho}{dx}+\frac{1}{T}\frac{dT}{dx}-\frac{1}{\overline{MW}}\frac{d\overline{MW}}{dx} \qquad \textit{Equation 3}$$

$$\frac{dMW}{dx}=-\overline{MW}^2\left(\sum_i\frac{1}{MW_i}\frac{dY_i}{dx}\right) \qquad \textit{Equation 4}$$

2.3.4 Species Conservation

As the fuel and air mixture travels downstream, a proportion of the fuel reacts with the air; this reaction process has previously been modelled using either equilibrium chemistry [1] [2] or chemical kinetics [5]. The equilibrium chemistry used in [2] considers fuel mixing then burning with air to produce products which are either set to the major species or are in thermodynamic equilibrium. This approach also assumes that the rate of change of the mass fraction for each of the i species present is determined by the rate of change of the local equivalence ratio, which is set by the fuel injection rate along the duct [2]. The alternate chemical kinetics approach presented in [5] references a derivation from [6]. The finite control volume analysis of species conservation, which neglects molecular diffusion in the flow direction, results in the differential species conservation equation presented in Equation 5.

$$\frac{dY_i}{dx}=\frac{\dot{\omega}_{i,mix}MW_i}{\rho U}+\frac{1}{\dot{m}}\frac{d\dot{m}_{i,added}}{dx}-\frac{Y_i}{\dot{m}}\frac{d\dot{m}}{dx} \qquad \textit{Equation 5}$$

This equation, from [5], models a situation in which the fuel is added to the system and mixes with the air. It is only once this mixing process is complete that reactions begin. The equation

would be more realistic if it described a situation in which reactions occur as the mixing progresses.

2.3.5 Conservation of Energy

Derivation of the energy equation in [5] follows a formulation presented in [6] and allows for the formulation of a differential equation for temperature. Consideration of a finite control volume, neglecting axial heat conduction, species diffusion, radiation and work leads to an equation for the conservation of energy, producing Equation 6 for the differential form of enthalpy.

$$\frac{dh}{dx} = \frac{1}{\dot{m}}\frac{d[\sum_i h_i \dot{m}_i]_{added}}{dx} - \frac{\dot{Q}'' P_w}{\dot{m}'' A} - \frac{h_0}{\dot{m}}\frac{d\dot{m}}{dx} - U\frac{dU}{dx} \qquad \textit{Equation 6}$$

The heat-transfer term was calculated from the definition of the Stanton number [5], expressed in Equation 7. This can be related to the friction coefficient using the Reynolds analogy [2], as expressed in Equation 8; this project has assumed a Prandtl number of 0.71, as given in [2] and [5]. Neither [2] nor [5] provide the value used for the friction coefficient, although [2] does direct the reader to an analytical technique presented in [7]. This technique presents a calculation method for c_f using the reference enthalpy method, and the Reynolds number evaluated at the reference temperature [7].

$$C_H = \frac{\dot{Q}''}{\rho U(h_{aw} - h_w)} \qquad \textit{Equation 7}$$

$$C_H = \frac{C_f}{2Pr^{\frac{2}{3}}} \qquad \textit{Equation 8}$$

It was assumed [5] that the wall enthalpies can be written as a function of wall and adiabatic wall temperatures and freestream specific heat at constant pressure, where adiabatic wall temperature can be calculated [7] using Equation 9; it was again assumed that the reference Prandtl number, Pr^*, is 0.71 [2].

$$T_{aw} = T\left[1 + (Pr^{*})^{\frac{1}{3}}\frac{M^2(\gamma-1)}{2}\right] \quad \textit{Equation 9}$$

Substitution of Equation 7, Equation 8 and Equation 9 into Equation 6 yielded [5] the energy equation, expressed as the differential form of temperature in Equation 10.

$$\frac{dT}{dx} = \frac{1}{\hat{c}_p}\left[-\sum_i h_i\frac{dY_i}{dx} + \frac{1}{\dot{m}}\sum_i\left(h_i\frac{d\dot{m}_i}{dx}\right)_{added} - \frac{2C_f c_p(T_{aw}-T_w)}{Pr^{\frac{2}{3}}\mathcal{D}A} - \frac{h_0}{\dot{m}}\frac{d\dot{m}}{dx} - U\frac{dU}{dx}\right] \quad \textit{Equation 10}$$

In this derivation [5], the specific heat variation of each species with temperature was described by Equation 11; from Equation 12, which describes the derivative of the specific heat, Equation 13 is produced, and in turn Equation 14.

$$c_{p_i} = \frac{R_u}{MW_i}\left(a_{1i} + a_{2i}T + a_{3i}T^2 + a_{4i}T^3 + a_{5i}T^4\right) \quad \textit{Equation 11}$$

$$\frac{dc_{p_i}}{dx} = \tilde{c}_{p_i}\frac{dT}{dx} \quad \textit{Equation 12}$$

$$\tilde{c}_{p_i} = \frac{R_u}{MW_i}\left(a_{2i} + 2a_{3i}T + 3a_{4i}T^2 + 4a_{5i}T^3\right) \quad \textit{Equation 13}$$

$$\hat{c}_p = \tilde{c}_p - \frac{1}{\dot{m}}\left\{\sum_i\left[\dot{m}_i\left(c_{p_i} + \tilde{c}_{p_i}T\right)\right]_{added}\right\} \quad \textit{Equation 14}$$

2.4 Solution Methodology

A stiff set of Ordinary Differential Equations (ODEs) are presented in Equation 1, Equation 2, Equation 3, Equation 4, Equation 5 and Equation 10. These were previously solved [5] using a stiff ODE solver known as VODPK [8], which used a backward-difference formula to integrate the stiff set of ODEs. CHEMKIN-II [9] was used to obtain values for the individual chemical species molecular weight, specific heat, heat of formation and reaction rates. The set of stiff ODEs were also solved to give Equation 15 for the velocity differential, supported by Equation 16 and Equation 17.

$$\frac{dU}{dx} = \frac{1}{\alpha}\Bigg\{-\frac{1}{A}\frac{dA}{dx} + \frac{1+\gamma M^2(1-\varepsilon)-\frac{h_0}{\hat{h}}}{\dot{m}}\frac{d\dot{m}}{dx}$$

$$+\frac{1}{\hat{h}}\left[-\sum_i h_i\frac{dY_i}{dx}+\frac{1}{\dot{m}}\sum_i\left(h_i\frac{d\dot{m}_i}{dx}\right)_{mix}\right]-\frac{1}{MW}\frac{dMW}{dx} \qquad \textit{Equation 15}$$

$$+\left[\gamma M^2-\frac{c_p(T_{aw}-T_w)}{\hat{h}Pr^{\frac{2}{3}}A}\right]\frac{2C_f}{\mathcal{D}}\Bigg\}$$

$$\alpha = \frac{1}{U}\left(1-\gamma M^2+\frac{U^2}{\hat{h}}\right) \qquad \textit{Equation 16}$$

$$\hat{h} = \hat{c}_p T \qquad \textit{Equation 17}$$

A solution methodology was presented in [5], using a user-defined cross-section area profile $\frac{dA}{dx}$ and mass mixing profile $\frac{d\dot{m}}{dx}$. The change in mixture molecular weight was then found by solving Equation 5 and substituting into Equation 4. It is then stated that the velocity derivative in Equation 15 can be solved, with any unknown friction terms first found using CHEMKIN. The density derivative could then be solved by rearranging Equation 1 followed by the pressure derivative using Equation 2, and finally the temperature derivative using Equation 3. Consideration of the properties required to determine the complete properties of the system shows that the method followed by [5] produces an over-defined system. The calculation of the temperature derivative was not necessary in each step, and the other properties could have been used to calculate the temperature at each point along the combustor length once the set of stiff ODEs had been solved.

Equation 1 and Equation 2 were rearranged in [2] to give explicit equations for the density and pressure derivatives, however an error was made in the equation for the pressure derivative, with a $'\rho'$ being in the place of a $'p'$ (Equation (26) of [2]). When Equation 2 was rearranged, the $\frac{1}{2U^2}\frac{dU^2}{dx}$ term was rearranged to $\frac{1}{U}\frac{dU}{dx}$ as there was no expression for the derivative of velocity

squared. Implementation of this method [5], integrating the derivatives of each variable using VODPK to find a flow solution, has allowed for rapid design of full vehicle concepts. This method has been described as computationally demanding [2], and the simple equilibrium chemistry method was continuously given preference as accurate results could be obtained with less computational expense.

2.5 Mass Mixing Profile

The mixing of fuel and air is typically calculated by solving Navier-Stokes equations in conjunction with a turbulence model [2]. However, this can be simplified by prescribing a fuel-mixing profile, which models when the injected fuel is available for reaction and can take the form of a mixing efficiency and/or a mixing length [5]. The mixing efficiency is defined as the ratio of fuel that is available for combustion to the amount that was injected [2]; whereas the mixing length describes the scenario where a finite amount of mixing time, after the fuel has been added to the flow, is necessary before reaction can occur [5]. In this later case, fuel is added to the flow over length L_{inj} and a prescribed mixing length, L_{mix}, is used to express how long the fuel must remain in the combustor before reaction can begin. The former case is used to model unmixed fuel in the combustor but also represents inefficiencies and losses as a result of fuel injection.

The mass mixing model [5] for perpendicular injection of hydrogen, is the curve fit of data [10] [11], expressed by Equation 18 and Equation 19; the derivative [5] of this is expressed in Equation 20. It was assumed that the injection mixing length is the same as the reaction mixing length [5].

$$\dot{m}_r = \dot{m}_f \frac{a\bar{x}^b e^{c\bar{x}}}{d\bar{x} + f} \qquad \textit{Equation 18}$$

$$\bar{x} = \frac{x}{L_{inj}} \qquad \textit{Equation 19}$$

$$\frac{d\dot{m}}{dx} = \frac{\dot{m}_r}{L_{inj}}\left[\frac{cd\bar{x}^2 + (bd + cf - d)\bar{x} + bf}{\bar{x}(d\bar{x} + f)}\right] \quad \textit{Equation 20}$$

When this model was compared [5] to experimental data, less than 15% average error was observed and the peak pressure location prediction compared favourably to the experimental data. It was noted that, compared to an equilibrium solution, the finite-rate chemistry also allowed for accurate prediction of fuel ignition [5]. However, the model does not account for boundary-layer burning or boundary-layer separation caused by injection, which was instead done by increasing the heat addition through setting of the injection length. Despite these limitations, the model rapidly predicted the pressure profile on the expansion surface, vital for thrust prediction, and accurately predicted the fuel ignition point [5].

The alternate method of using a mixing efficiency to model the fuel available for reaction [2] uses Equation 21, with the mixing efficiency changing in value from zero at the injector exit until reaching near unity at the defined mixing length.

$$\dot{m}_f = \eta_m \dot{m}_{f0} \quad \textit{Equation 21}$$

2.6 Specific Heats

The derivation in [5] based the specific heat variation with temperature given by Equation 11, and the analysis lead to the specific heat terms given in Equation 12, Equation 13 and Equation 14. Basing the specific heat variation with temperature instead on [12] gives Equation 25. Following the analysis of [5], assuming a Hydrogen fuelled scramjet, results in Equation 26 and Equation 27. The coefficients a_{1i} through to a_{7i} were obtained from [12], as were the enthalpies of standard state (0 K); these values are available in Appendix B.

$$c_{p_i} = \frac{R_u}{MW_i}\left(a_{1i}T^{-2} + a_{2i}T^{-1} + a_{3i} + a_{4i}T + a_{5i}T^2 + a_{6i}T^3 + a_{7i}T^4\right) \quad \textit{Equation 22}$$

$$\tilde{c}_{p_i} = \frac{R_u}{MW_i}\left(-2a_{1i}T^{-3} - a_{2i}T^2 + a_{4i} + 2a_{5i}T + 3a_{6i}T^2 + 4a_{7i}T^3\right) \quad \textit{Equation 23}$$

$$\hat{c}_p = \tilde{c}_p - \frac{\dot{m}_f}{\dot{m}}\left(c_{p_{H_2}} + \tilde{c}_{p_{H_2}} T\right)$$ *Equation 24*

2.7 Chemical Kinetics

The use of chemical kinetics to describe the reactions occurring inside the combustor allows for a prediction of fuel ignition, a finite rate process which inherently cannot be predicted using equilibrium methods [5]. CHEMKIN [9] has previously been used for this [5] [13] as the chemical kinetics for chain branching and propagating mechanisms can be quite complicated. Multistep reaction mechanisms are presented in [6], with documented values for the unknowns in the Arrhenius equation, Equation 25. Whereas a simplified single step reaction, described by Equation 26, is developed and validated in [13] and given in Equation 27, with $\frac{E_a}{R_u}$ given in Kelvin and A in mole/(cm^{3}s). This model was validated against data derived from flame propagation and structure, and shock tube ignition delay studies, though the single step reaction is not as accurate as multiple step reaction mechanisms.

$$\dot{\omega} = Ae^{-\frac{E_a}{R_u}T}[A]^m[B]^n$$ *Equation 25*

$$H_2 + \frac{1}{2}O_2 \rightarrow H_2O$$ *Equation 26*

$$\dot{\omega} = 1.8\times10^{13}e^{\left(-\frac{17614}{T}\right)}[H_2]^{1.0}[O_2]^{0.5}$$ *Equation 27*

An alternative to explicitly using the Arrhenius equation is to utilise an open source multi-physical simulation software, Eilmer [14]. This simulation code can be used to calculate the reaction rates from given mass fractions, temperature and pressure. Utilisation of this software allows for more complex multiple step reaction mechanisms to be implemented. [15] presents a nine-species, 18-reaction scheme, which includes $O,\ O_2, N_2, H, H_2, H_2O, OH_2$, and H_2O_2. The reactions are given in Appendix C.

3.0 Solution Methodology

For known constants, initial conditions, area profile ($\frac{dA}{dx}$) and mass mixing profile ($\frac{d\dot{m}}{dx}$), a set of ODEs (Ordinary Differential Equations) can be solved at each step along the combustor to determine flow properties. The Python code developed is included in Appendix D.

3.1 Differential Values

The velocity, density and pressure of the flow, along with the mass fraction of each species are the inputs to the ODEs. The temperature of the flow is then calculated (Equation 28), followed by the individual specific heats of each species (Equation 22, Equation 23 and Equation 29). Once these are known, the overall specific heats, ratio of specific heats and mixture molecular weight are calculated (Equation 24, Equation 30 and Equation 31). From here the Mach number of the flow is determined (Equation 32). The enthalpy of each species is also calculated (Equation 33), as is the adiabatic wall temperature (Equation 9). Then, with these parameters known, as well as the area, differential of area, mass flow rate (Equation 34), and differential of mass flow rate, the calculation of the ODE terms can proceed.

$$T = \frac{p}{\rho R} \qquad \textit{Equation 28}$$

$$c_{v_i} = c_{p_i} - \frac{R_u}{MW_i} \qquad \textit{Equation 29}$$

$$c_{p/v} = \sum_i c_{p_i/v_i} Y_i \qquad \textit{Equation 30}$$

$$\overline{MW} = \frac{1}{\sum_i \frac{Y_i}{MW_i}} \qquad \textit{Equation 31}$$

$$M = \frac{U}{\sqrt{\frac{\gamma p}{\rho}}} \qquad \textit{Equation 32}$$

$$h_i = h_{ref_i} + C_{p_i}(T - T_{ref}) \quad \textit{Equation 33}$$

$$\dot{m} = \rho U A \quad \textit{Equation 34}$$

The reaction rates are calculated through use of Eilmer. Appendix E contains a code [16] which calls Eilmer gas and chemistry models to compute reaction rates for species. This code makes use of the gas model code [17] contained in Appendix F, and the chemical kinetic code [18], which corresponds to [15], contained in Appendix G.

The differential of mass fraction for each of the species can then be determined. Equation 35 specifies an adjusted form of Equation 5, which allows the simultaneous mixing and reacting of the flow.

$$\frac{dY_i}{dx} = \frac{\dot{\omega}_i MW_i}{\rho U} + \frac{1}{\dot{m}}\frac{d\dot{m}_{i,mix}}{dx} - \frac{Y_i}{\dot{m}}\frac{d\dot{m}}{dx} \quad \textit{Equation 35}$$

Following this, the differential mixture molecular weight could be determined using Equation 4. The velocity differential was then evaluated from Equation 15, making use of Equation 16 and Equation 17. Equation 1 was rearranged to give Equation 36, which is used to determine the density differential. The pressure differential can be determined from Equation 37, which is a rearranged form of Equation 2.

$$\frac{d\rho}{dx} = \rho\left(\frac{1}{\dot{m}}\frac{d\dot{m}}{dx} - \frac{1}{U}\frac{dU}{dx} - \frac{1}{A}\frac{dA}{dx}\right) \quad \textit{Equation 36}$$

$$\frac{dp}{dx} = -\gamma p M^2\left(\frac{1}{U}\frac{dU}{dx} + \frac{2C_f}{\mathcal{D}} + \frac{1}{\dot{m}}\frac{d\dot{m}}{dx}\right) \quad \textit{Equation 37}$$

This set of Equations was sufficient to determine the parameters, $\frac{dU}{dx}$, $\frac{d\rho}{dx}$, $\frac{dp}{dx}$, $\frac{dY_O}{dx}$, $\frac{dY_{O_2}}{dx}$, $\frac{dY_{N_2}}{dx}$, $\frac{dY_H}{dx}$, $\frac{dY_{H_2}}{dx}$, $\frac{dY_{H_2O}}{dx}$, $\frac{dY_{HO_2}}{dx}$, $\frac{dY_{OH}}{dx}$ and $\frac{dY_{H_2O_2}}{dx}$, necessary to describe the flow field at each point along the length of the combustor. Utilisation of an ODE solver thus gave values for U, ρ, p, Y_O, Y_{O_2}, Y_{N_2}, Y_H, Y_{H_2}, Y_{H_2O}, Y_{HO_2}, Y_{OH} and $Y_{H_2O_2}$ at each point along the length of the combustor.

3.2 Flow Properties

The known flow properties could then be used to determine other parameters at each position. The temperature (Equation 28), specific heats (Equation 22, Equation 23, Equation 29 and Equation 30), mixture molecular weight (Equation 31), Mach number (Equation 32), mass flow rate (Equation 34), and species enthalpy values (Equation 33) were calculated. The stagnation enthalpy, given by Equation 38, was also calculated to allow confirmation of the results with known stagnation enthalpy trends.

$$h_0 = \frac{U^2}{2} + \sum_i h_i Y_i \qquad \textit{Equation 38}$$

3.3 ODE Solvers

Initially, MATLAB was used to develop the computational method with the single step reaction mechanism described by Equation 26. The stiff ODE solver [19] chosen was originally ode23s; however, the mass matrix was close to singular so this was changed to ode15s, which is a slightly less efficient solver. This provided results for the case with no mass addition and for the one with mass addition and mixing but no reactions. However, using Equation 27 to calculate the reaction rate for this simplified chemistry, and utilising the results from this presented errors. Thus, the more complex 18-reaction scheme [15] was to be implemented as it would provide more realistic results.

Implementation of this scheme [15] would require the use of Eilmer [14]. Eilmer requires a Linux operating system and the MATLAB functions would need to import Python codes. Attempts to implement this proved unsuccessful; so, the computational method was rewritten using Python. Within Python the scipy.integrate.odeint function [20] was chosen to integrate the set of ODEs as it requires the input of initial values.

3.4 Geometry and Mixing Profile

For the development of the code, combustor geometry and initial conditions were set to match the first example in [5], of a Hydrogen fuelled scramjet. This was done for simplicity and allowed quick comparison of results. The geometry of this combustor was an initial area of $0.0038\ m^2$ for 0.28 m, followed by a constant expansion to twice this area, over a length of 0.61 m. The initial conditions were: $U = 1849\ m/s$, $p = 52\ kPa$, $T = 872\ K$, $T_w = 500\ K$ and $M = 3.19$. The mass mixing profile used for this was that described by Equation 18, Equation 19 and Equation 20, though an additional $10^{-14}\ m$ was added to the $\bar{x}$ calculation to ensure the differential did not go to infinity at the start of the combustor.

Once the code was producing reasonable results, the geometry and mixing profile were adjusted to match the Mach 12 REST engine [21]. The geometry of this engine is shown in Figure 5, and the experimental mass mixing profile in Figure 6. For this analysis, it was considered that the fuel was injected at the start of the combustor, that there is no isolator, and that there is thus no step or turn in the profile. The initial geometry of the combustor was an ellipse with a width of 31.05 mm and an aspect ratio of 1.76 [21], giving an area of $0.001721\ m^2$. This area is constant for 215.67 mm, after which it diverges to twice the initial area, over a length of 121 mm.

The initial combustor conditions, without fuel injection in the inlet of the engine, are $\rho = 0.1023\ kg/m^3$, $T = 1560\ K$, $p = 44.502\ kPa$, $U = 3027\ m/s$ and $M = 4.02$ [22]. Though the real engine includes fuel injection in the inlet, and this changes the combustor inlet conditions, the case being considered here is without this fuel addition.

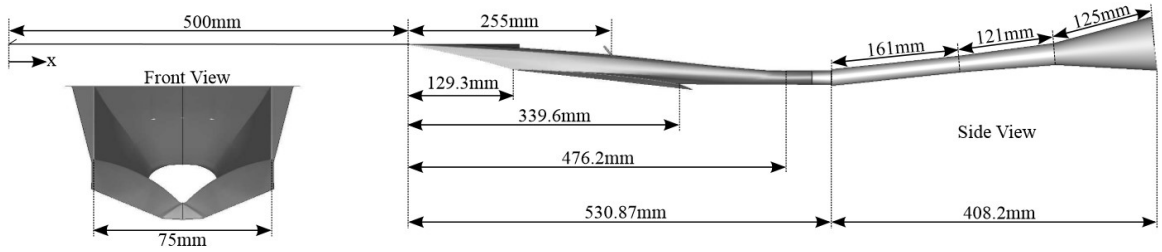

Figure 5: M12 REST Engine flow path geometry [21]

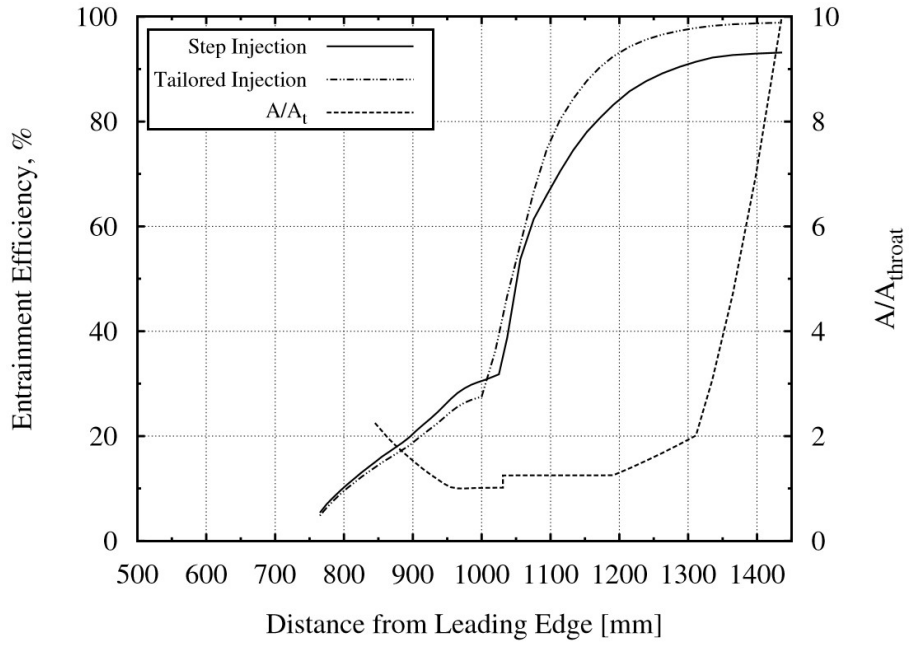

Figure 6: Mixing Efficiency [21]

The mass mixing profile was adapted from the plot in Figure 6: Mixing Efficiency Figure 6, considering the tailored efficiency curve for the entrainment (mixing) efficiency. The maximum efficiency was 98.9%, so this will be the value at the end of the combustor. The initial value, at the start of the combustor, would be zero, as the case being modelled is without inlet fuelling. A curve was fitted to points from this case as shown in Figure 7. This produced an exponential function, of the form given in Equation 39, with constants a, b, c and d.

$$\eta_m = ae^{bx} + ce^{dx} = 1.168e^{-0.3605x} - 1.182e^{-9.525x} \quad \textit{Equation 39}$$

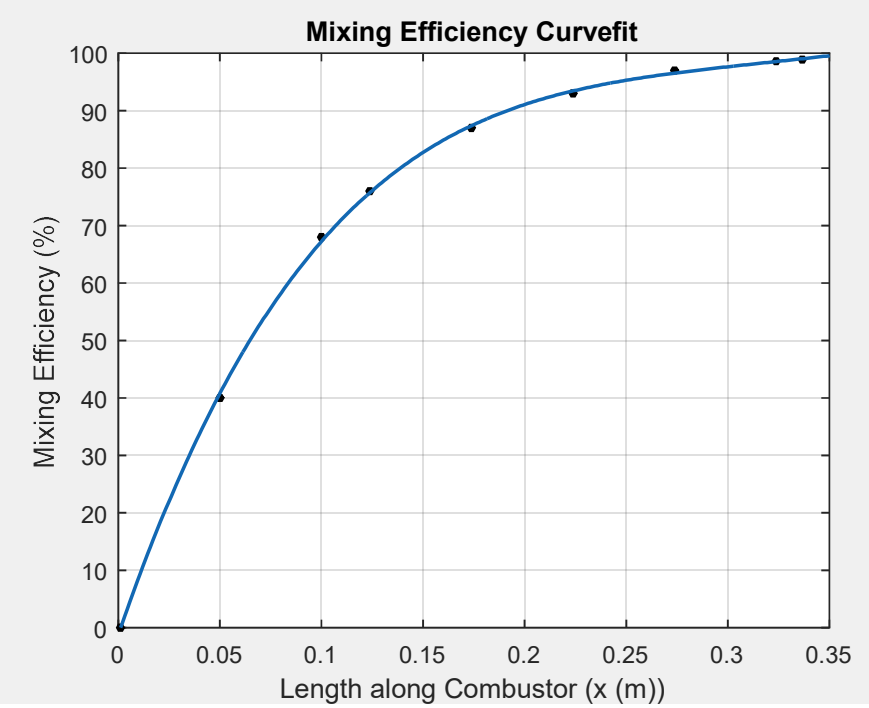

Figure 7: Mixing Efficiency Curve-fit

For the prescribed length of the M12 REST engine, this curve-fit describes the realistic case where the fuel is not fully mixed with the air-stream by the end of the combustor. The mass flow rate differential will then be given by Equation 40, which is derived from Equation 21, along with Equation 41.

$$\frac{d\dot{m}}{dx} = \frac{d\dot{m}_f}{dx} = \frac{d\eta_m}{dx}\dot{m}_{f0} \qquad \textit{Equation 40}$$

$$\frac{d\eta_m}{dx} = abe^{bx} + cde^{dx} = -0.4210e^{-0.3605x} + 11.26e^{-9.525x} \qquad \textit{Equation 41}$$

4.0 Results

The Mach 12 REST engine geometry was used as a basis for the investigation of how geometric changes affect combustor properties. The initial conditions used were $\rho = 0.1023\ kg/m^3, T = 1560\ K, p = 44.502\ kPa, U = 3027\ m/s$ and $M = 4.02$ [22]. The mass flow rate differential used was the curve-fit given by Equation 40 and Equation 41. The equivalence ratio was set to 0.75.

The properties at the end of the combustor were recorded, along with the geometry specified to produce them. The combustion efficiency can be examined by considering the mass fraction of water, Y_{H_20}, at the end of the combustor; the higher this value, the more complete the reactions. Plots were created of the flow properties along the combustor. As the mass fraction of Nitrogen is approximately four times the other mass fraction values, it was plotted at one quarter of its value so that the other mass fraction values would be more clear.

4.1 Mach 12 REST Case

The first case considered was the Mach 12 REST geometry; an initial area of 0.001721 m^2 which continues for 0.21567 m, before diverging over 0.121 m to a final area of 0.003442 m^2. This was first examined without friction or heat losses to the walls of the engine. The results of this simulation are displayed in Figure 8, with the final values contained in Table 1. The case with friction was then considered, with the results contained in Figure 9 and final values in Table 1. The coefficient of friction, C_f, value was set to 2×10^{-3}.

The results in Table 1, along with the plots in Figure 8 and Figure 9, show that the inclusion of friction decreases the velocity of the flow, and increases the density, pressure and temperature. This results in a decrease in Mach number and in the combustion efficiency of the engine. Examining the species mass fractions, it can be noted that, in addition to the H_2O, there is a noticeable amount of OH, though not a large amount of the other species containing Hydrogen. This would limit the amount of H_2O produced in the combustor, thus limiting the efficiency of

the combustion. This indicates that the reactions do not all go to completion. Another noticeable characteristic is the time it takes for ignition to occur. This can be seen in the initial section of the combustor where Hydrogen begins to build up before combustion is initiated. The Hydrogen mass fraction then decreases rapidly as the reactions proceed and the mass fraction of water increases. A small bump in the other values can be seen when the combustion is initiated; the pressure and temperature of the flow are driven upwards at an increased rate, which further increases the Mach number of the flow.

Problems were encountered when heat loss to the walls was incorporated into the computation, through use of Equation 9. Inclusion of this term decreased the temperature of the flow to below 200 K (the limit of the specific heat curve-fits used [12]). This would effectively choke the flow as the Mach number decreased along with other flow properties. It is also well below a realistic temperature range which would be expected inside the combustor. Thus, the heat loss to the walls of the engine were not considered for further analyses; this was carried out by setting the adiabatic wall temperature, T_{aw}, equal to the wall temperature, T_w. This removes the relevant term from Equation 15.

Table 1: Mach 12 REST Engine Results

	Initial Area, A_1 ($m^2 \times 1000$)	**Final Area, A_2 ($m^2 \times 1000$)**	**Constant Area Length, L_1 (m)**	**Diverging Area Length, L_2 (m)**	**Final Velocity, U (m/s)**	**Final Density, ρ (kg/m^3)**	**Final Pressure, p (kPa)**	**Final Temperature, T (K)**	**Final Mach Number, M**	**Final Mass Fraction of Water, Y_{H_2O}**
Frictionless Case	1.721	3.442	0.216	0.121	3051	0.052	36.22	2420	3.22	0.1195
Friction Case	1.721	3.442	0.216	0.121	2967	0.053	39.88	2591	3.03	0.1158

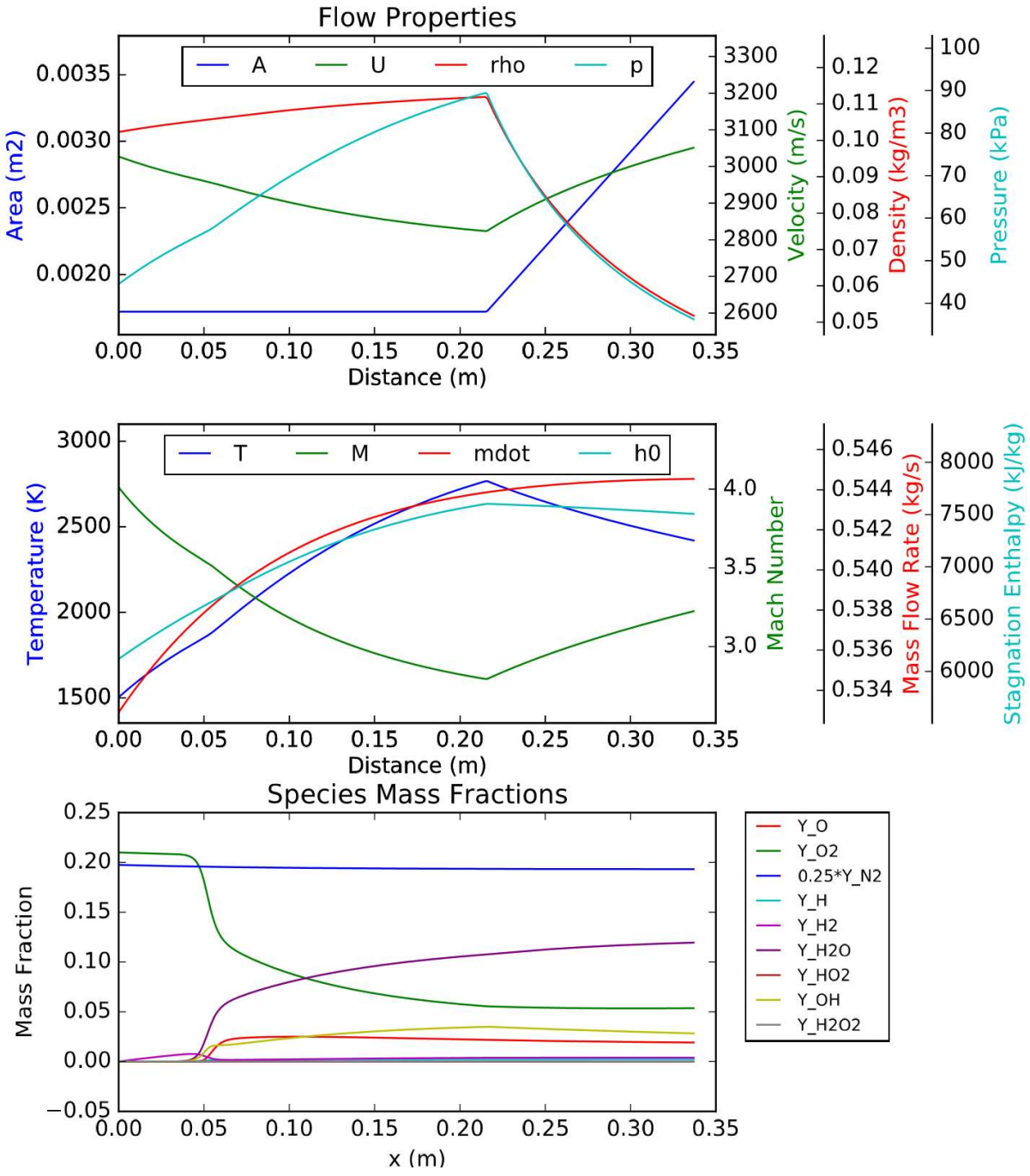

Figure 8: Mach 12 REST Engine Geometry Frictionless Results

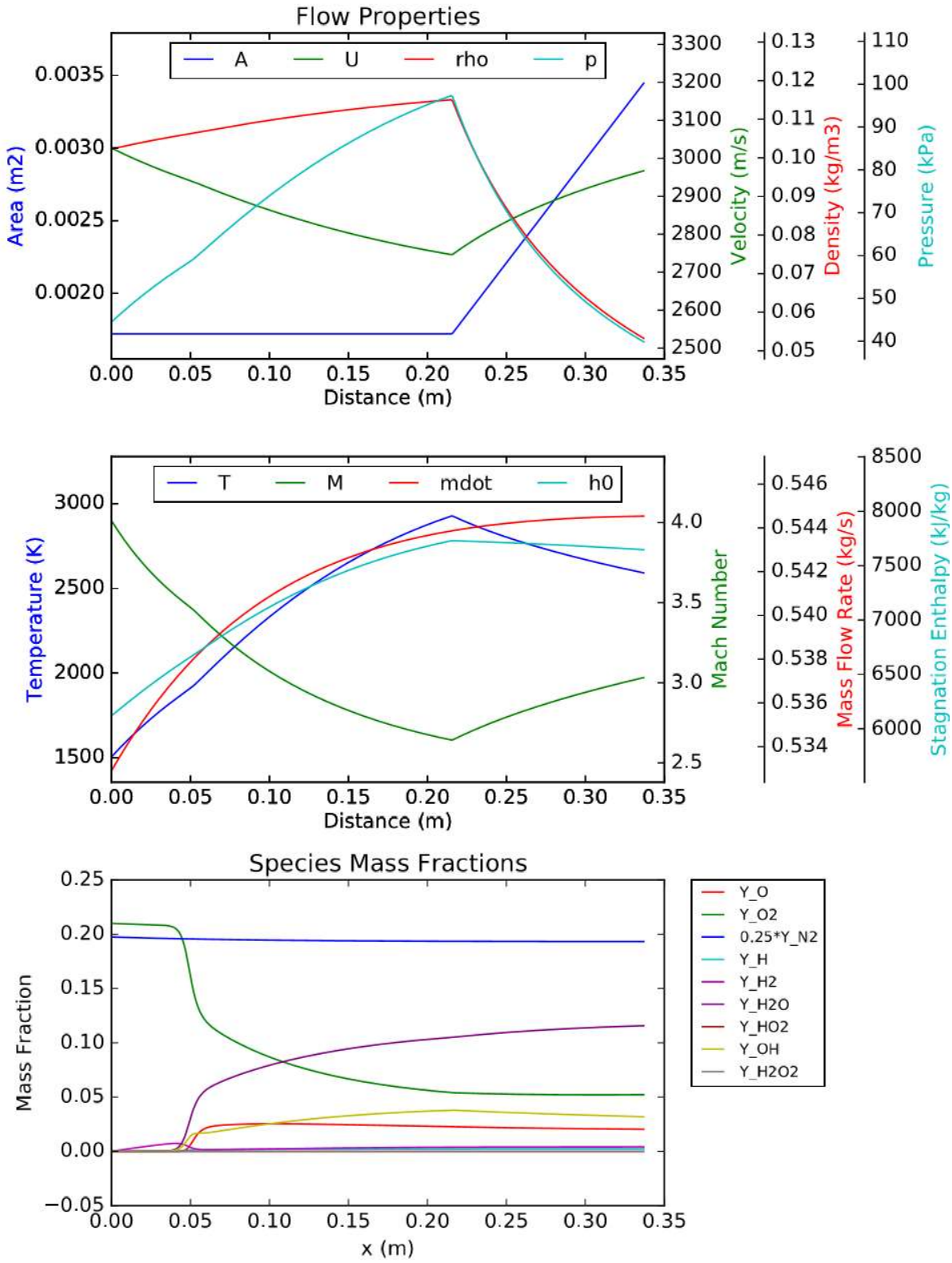

Figure 9: Mach 12 REST Engine Geometry Friction Results

4.2 Constant Area Combustor

Further examination of the variance of the combustor's flow properties and combustion efficiency, with respect to changes in geometry was carried out. For these analyses, friction losses were again included, though heat losses to the combustor walls were not. The total length and possible areas remained the same as the M12 REST case, though the point at which the combustor began to diverge was adjusted to examine the response.

The first case considered was that of a constant area combustor. The results for this are displayed in Figure 10. Table 2 contains the flow properties at the end of the combustor. This shows that the flow within the constant area combustor has a lower velocity, and higher density, pressure and temperature than the Mach 12 REST case. This decreases the Mach number and the combustion efficiency of the engine. It can be observed that this trend in values is similar to that between the frictionless and friction case examined earlier, though the values vary more significantly.

Comparison of Figure 9 and Figure 10 highlights that when the area of the combustor is constant the velocity is driven down and the density, pressure and temperature are driven up. When the combustor area begins to diverge, a sharp change can be noted and the velocity begins to increase while the density, pressure and temperature decrease. It is in this diverging section that the OH mass fraction decreases slightly, allowing a further increase in the mass fraction of water, and thus in the combustion efficiency. The constant area engine does not share this characteristic; instead the OH mass fraction continues to increase throughout the combustor, limiting the combustion efficiency. Interestingly, no dissociation of the water molecules can be observed. It was expected that for the constant area combustor the high temperatures and pressures would result in the mass fraction of water beginning to decrease towards the end of the combustor.

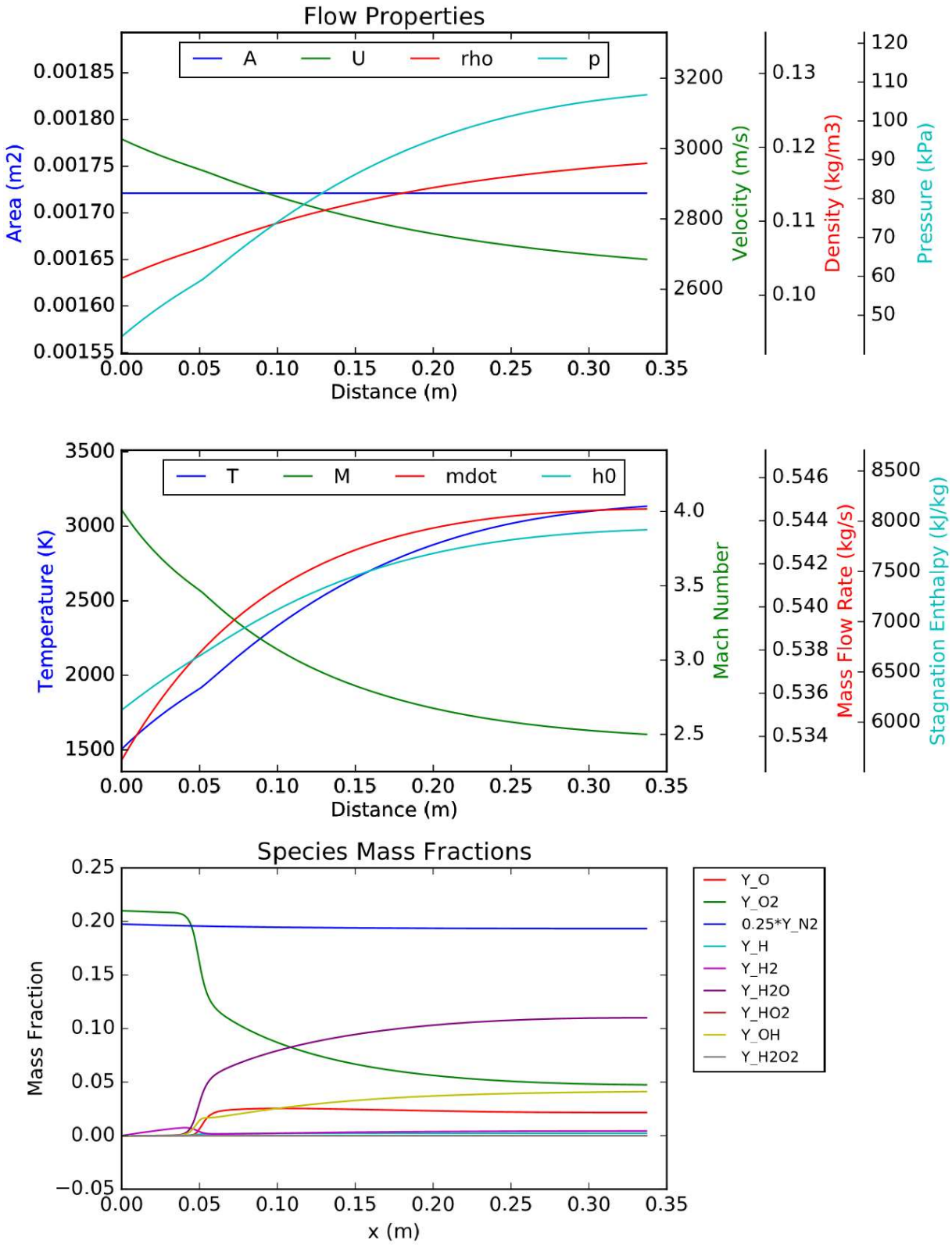

Figure 10: Constant Area Combustor Results

4.3 Constantly Diverging Combustor

The case of a constantly diverging combustor was next considered. The initial and final areas of the combustor were set the same as those in the Mach 12 REST engine, though the area was set to vary at a constant rate over the entire length of the combustor. The results are displayed in Figure 11, and the properties at the end of the combustor area included in Table 2.

Comparing the values at the end of the combustor to those of Mach 12 REST engine, it can be seen that the velocity is slightly lower, as is the pressure and temperature, and that the density is slightly higher. This gives a slightly higher Mach number, however the mass fraction of water, and thus the combustion efficiency, is lower. Comparison with the constant area case shows that the constantly diverging case has higher velocity, and lower density, pressure and temperature. This gives the constantly diverging case a higher Mach number and greater combustion efficiency. It is notable that the constantly diverging case has values which are more similar to the Mach 12 REST case than the constant area case; this is most noticeable in the Mach numbers. However, this is not reflected in the combustion efficiency, with the constantly diverging area case marking an approximate half-way between the constant area case and the Mach 12 REST engine.

Comparison of Figure 9 and Figure 11 highlights the differences in flow property trends along the combustor. Noticeably, the density decreases gradually along the entire combustor length for the constantly diverging case; whereas, for the Mach 12 REST engine, it increases during the constant area section and then decreases rapidly during the diverging section. This suggests that the rate at which area changes has a direct effect on the rate at which density changes, as is supported by Equation 36; though interestingly the final values are highly similar. The trend shown by the velocity is also interesting; notably it decreases slowly in the constantly diverging combustor, compared to the increase observed during the diverging section of the Mach 12 REST engine.

The pressure also exhibits an interesting trend, with the initial increase plateauing before ignition causes further increase, this is then followed by a continuous decrease. The Mach 12 REST engine pressure increases significantly during the constant area section before a rapid decrease when the area begins to diverge. This again suggests that the rate at which the area changes directly affects the rate at which pressure changes, though the effect of additional parameters can be seen more clearly here; this is consistent with Equation 37.

The temperature decreases along the diverging section of the Mach 12 REST engine, a trend which is driven by the decreases in pressure and density. However, within the constantly diverging combustor, the initial increases in pressure and the gradual decrease in density, and later in pressure, can be seen to initially increase the temperature before it begins to plateau. Further, it should be noted that the ignition in the constantly diverging case occurs slightly later than the ignition in the Mach 12 REST engine, due to the slower increase in temperature. There is also no noticeable decrease in the mass fraction of OH, which is observed to occur when the area begins to diverge in the Mach 12 REST engine.

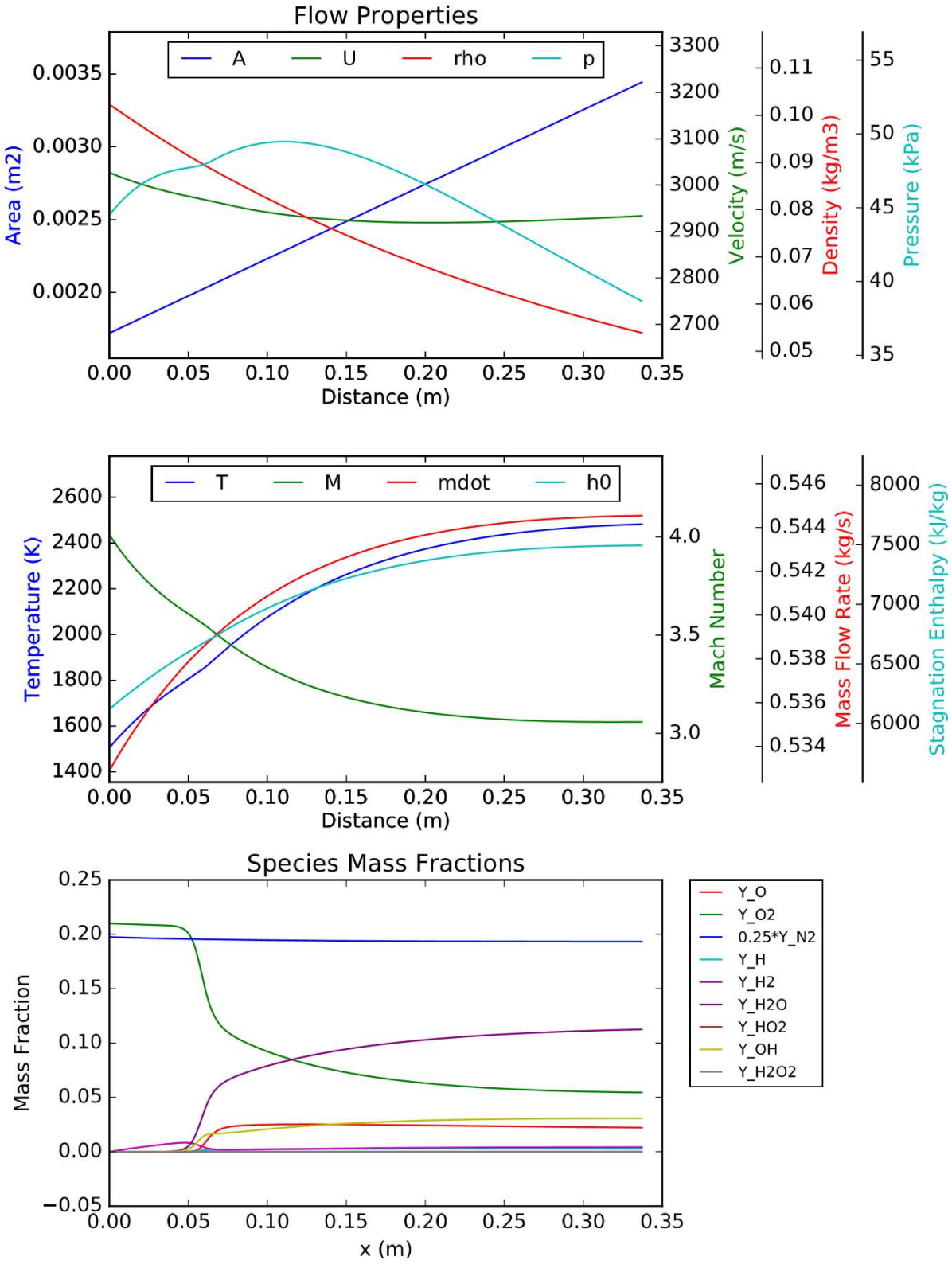

Figure 11: Constantly Diverging Combustor Results

4.4 Variation in Location of Divergence

The response of the flow properties to variations in the location along the combustor at which the area begins to diverge was next examined. It was found that if this distance decreased, giving a more gradual divergence which begins sooner, then the final mass fraction of water, and thus the combustion efficiency, would be decreased. However, the response to the distance being increased, giving a steeper divergence which begins later, was more varied. Increasing the distance increased the final mass fraction of water, up until a certain distance. This allowed the distance at which the divergence began to be varied until a maximum final mass fraction of water was found. The results for this were recorded, and are contained in Table 2. The initial and final areas of the combustor, as well as its total length were held consistent with the Mach 12 REST engine case.

Considering Table 2, the properties at the end of the combustor can be analysed with respect to the variation of the beginning of the diverging section. The final output velocity was found to decrease for cases where the divergence began sooner. With the exception of the constant area case, the density at the end of the combustor was found to be largely unaffected by changes in where the combustor began to diverge. The pressure and temperature increase when the divergence began closer to the start of the combustor, while the Mach number decreases.

These changes accumulate to vary the final mass fraction of water in the combustor, which is used as a measure of combustion efficiency. The order at which the mass fraction varies is at 10^{-3} and lower, thus values have been included to the order of 10^{-6} to allow comparison. The largest mass fraction of water produced was when the combustor began to diverge 0.231 m along its length (Variation 5). The results which pertain to this distance are displayed in Figure 12. These results appear very similar to those in Figure 9, as would be expected. However, it should be noted that the mass fraction of OH here is slightly lower, which gives the greater combustion efficiency as more reactions have gone to completion.

Table 2: Mach 12 REST - Variation of Divergence Location

	Initial Area, A_1 ($m^2 \times 1000$)	**Final Area, A_2 ($m^2 \times 1000$)**	**Constant Area Length, L_1 (m)**	**Diverging Area Length, L_2 (m)**	**Final Velocity, U (m/s)**	**Final Density, ρ (kg/m^3)**	**Final Pressure, p (kPa)**	**Final Temperature, T (K)**	**Final Mach Number, M**	**Final Mass Fraction of Water, Y_{H_2O}**
M12 REST	1.721	3.442	0.216	0.121	2967	0.053	39.88	2591	3.03	0.115804
Constant Area	1.721	1.721	0.337	0.0	2685	0.118	106.7	3134	2.50	0.110053
Constantly Diverging Area	1.721	3.442	0.0	0.337	2934	0.054	38.67	2482	3.06	0.112517
M12 REST Variation 1	1.721	3.442	0.316	0.021	2976	0.053	39.00	2558	3.06	0.115049
M12 REST Variation 2	1.721	3.442	0.266	0.071	2971	0.053	39.66	2583	3.04	0.115711
M12 REST Variation 3	1.721	3.442	0.241	0.096	2969	0.053	39.80	2589	3.04	0.115828
M12 REST Variation 4	1.721	3.442	0.236	0.101	2969	0.053	39.82	2590	3.04	0.115834
M12 REST Variation 5	1.721	3.442	0.231	0.106	2968	0.053	39.84	2590	3.03	0.115835
M12 REST Variation 6	1.721	3.442	0.226	0.111	2968	0.053	39.85	2591	3.03	0.115830

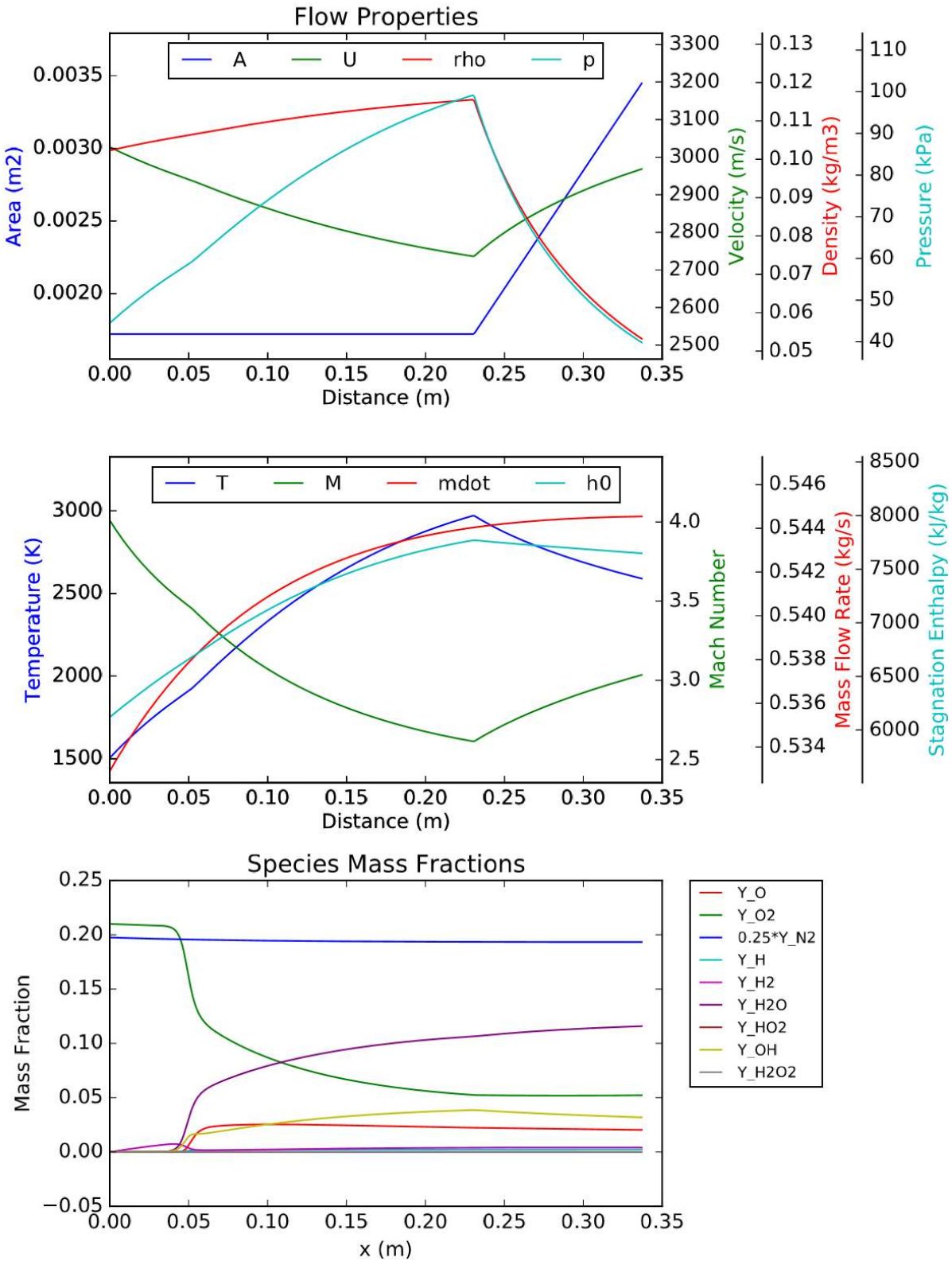

Figure 12: Variation on Mach 12 REST Engine - Divergence begins 0.015 m later

4.5 Variation in Magnitude of Divergence

Further investigation was conducted into how the flow properties are effected by variation in the final area to which the combustor diverges. This was done utilising the adjusted Mach 12 REST engine geometry from the investigation above. The initial combustor conditions and mass mixing profile remained consistent with the previous investigations. The final area was varied from 1.5 times the initial area to 2.5 times the initial area, with the properties at the end of the combustors displayed in Table 3. The results from each of these variations are also displayed in Figure 13, Figure 14, Figure 15 and Figure 16.

Table 3 indicates that the properties at the end of the combustor are dependent on the amount by which the combustor area diverges. Larger divergence increases the velocity, and decreases the density, pressure and temperature. This increases the Mach number and the mass fraction of water produced. Thus, greater divergence results in greater combustion efficiency.

Considering Figure 13, Figure 14, Figure 15 and Figure 16, it is clear that the rate at which the density, pressure and temperature decrease is greatly affected by the rate at which the area increases. This greater rate of decrease in density, pressure and temperature, causes the mass fraction of OH to decrease and the mass fraction of H_2O to increase.

While this effect of greater divergence is useful, there should exist a limit to how much expansion the flow can undergo without beginning to decrease these values too far. The degree of divergence which is reasonable should be considered further, with analysis into the flow behaviour around the transition and within the diverging section.

Calculations of the thrust which would be expected to be generated should also be considered for all cases here. This can be determined using the flow properties at the end of the combustor and expanding each case to the same area over the same length. This would equalise results and take losses, such as friction, into account, allowing for further comparison and optimisation.

Table 3: Mach 12 REST Engine - Variation of Final Area

	Initial Area, A_1 ($m^2 \times 1000$)	**Final Area, A_2 ($m^2 \times 1000$)**	**Constant Area Length, L_1 (m)**	**Diverging Area Length, L_2 (m)**	**Final Velocity, U (m/s)**	**Final Density, ρ (kg/m^3)**	**Final Pressure, p (kPa)**	**Final Temperature, T (K)**	**Final Mach Number, M**	**Final Mass Fraction of Water, Y_{H_2O}**
M12 REST Variation 5	1.721	3.442 ($2A_1$)	0.231	0.106	2968	0.053	39.84	2590	3.03	0.115835
M12 REST Variation 7	1.721	2.581 ($1.5A_1$)	0.231	0.106	2861	0.074	59.93	2816	2.81	0.114088
M12 REST Variation 8	1.721	3.012 ($1.75A_1$)	0.231	0.106	2920	0.062	48.13	2694	2.93	0.115079
M12 REST Variation 9	1.721	3.872 ($2.25A_1$)	0.231	0.106	3008	0.047	33.75	2502	3.13	0.116460
M12 REST Variation 10	1.721	4.302 ($2.5A_1$)	0.231	0.106	3042	0.042	29.11	2426	3.21	0.116999

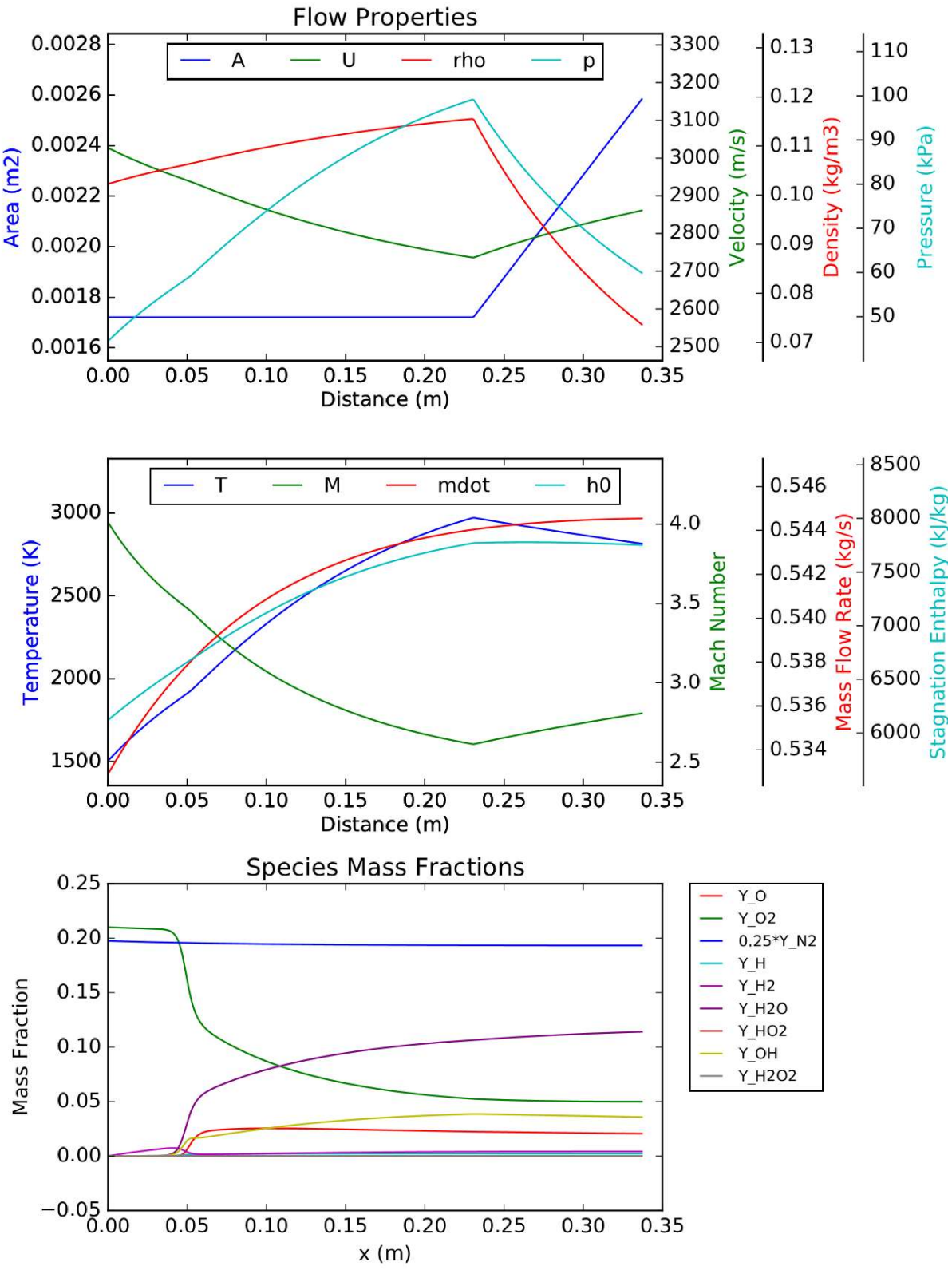

Figure 13 Mach 12 REST Engine Variation - $A_2 = 1.5A_1$

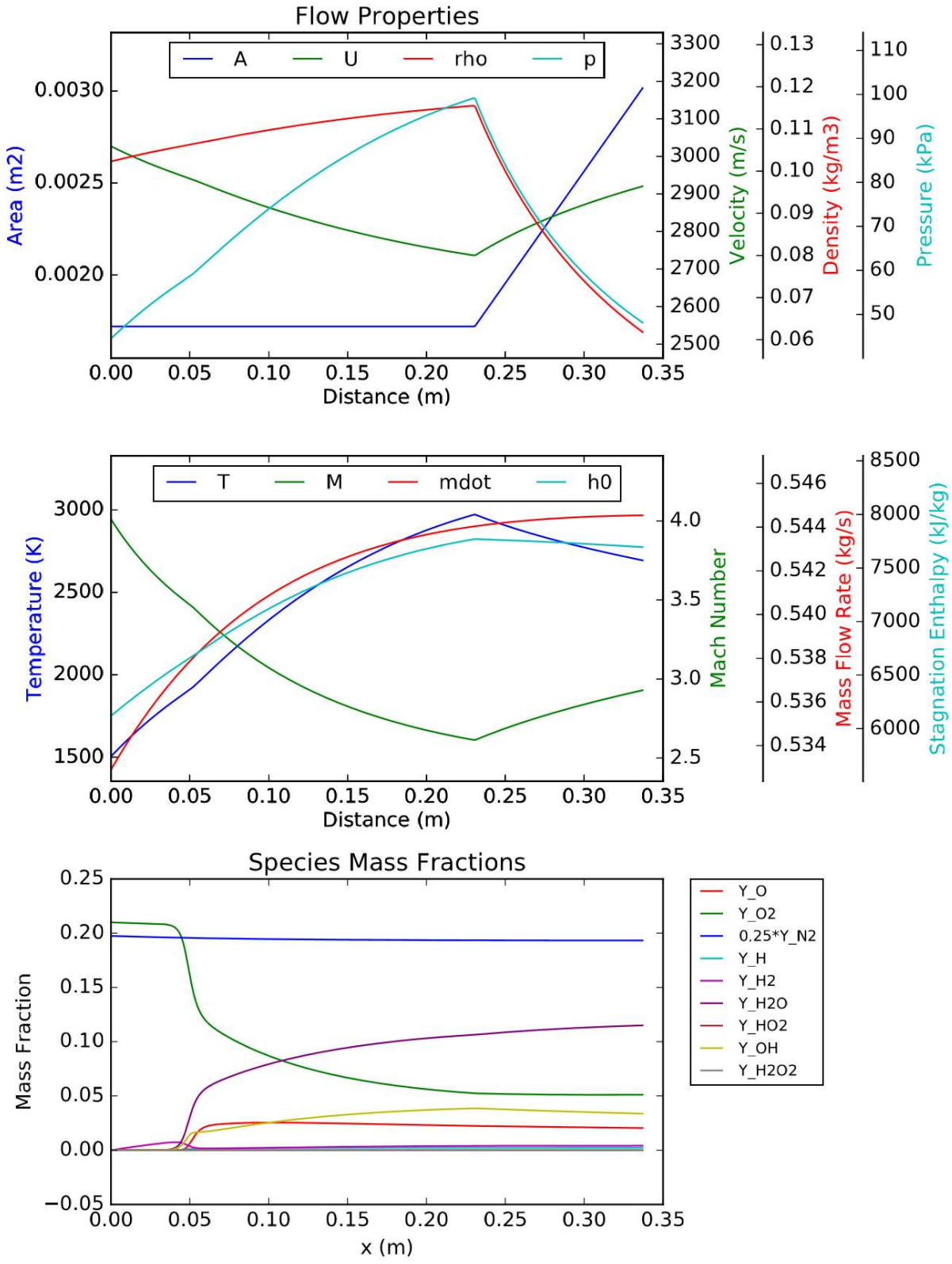

Figure 14: Mach 12 REST Engine Variation - $A_2 = 1.75A_1$

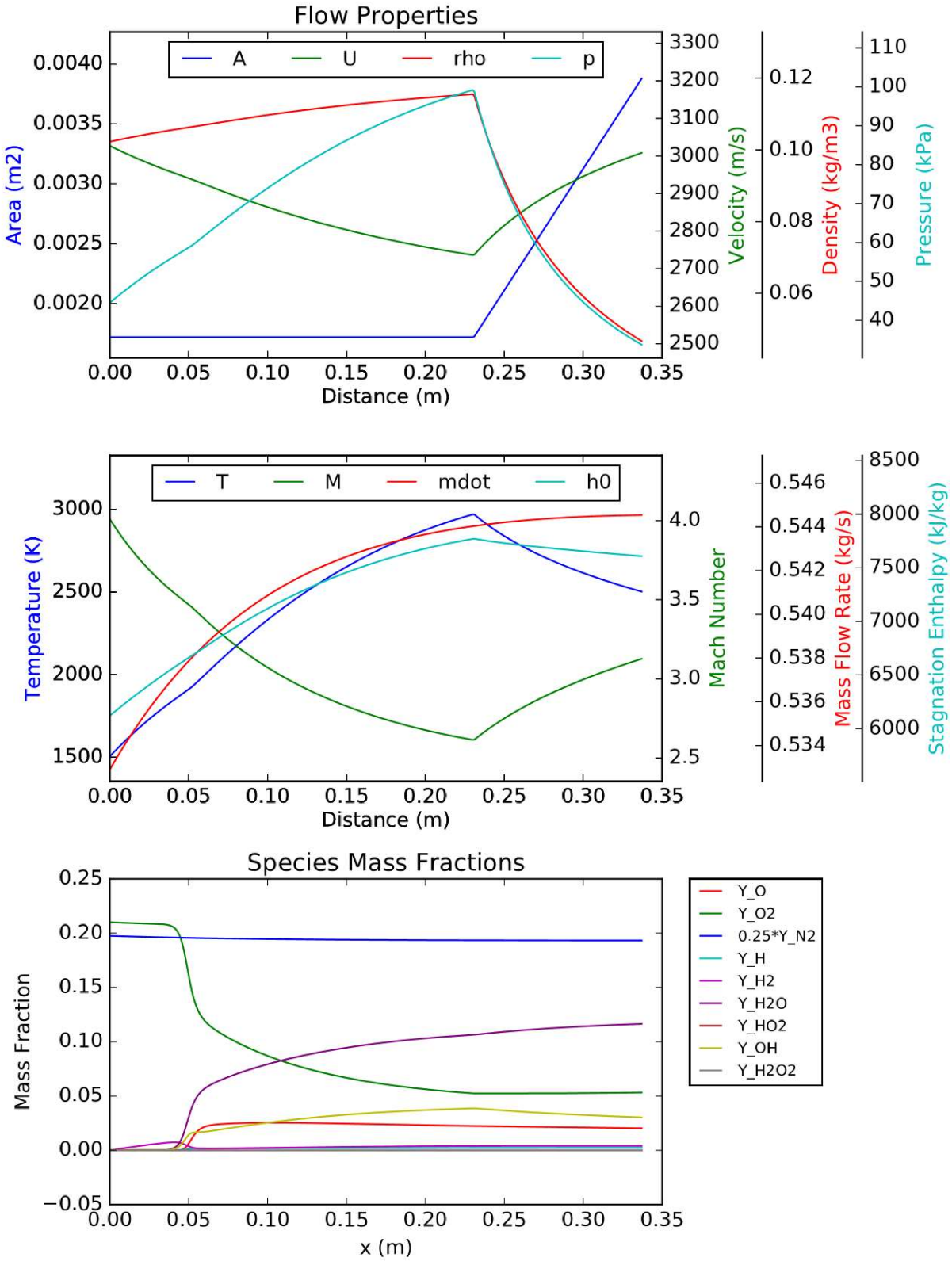

Figure 15: Mach 12 REST Engine Variation - $A_2 = 2.25A_1$

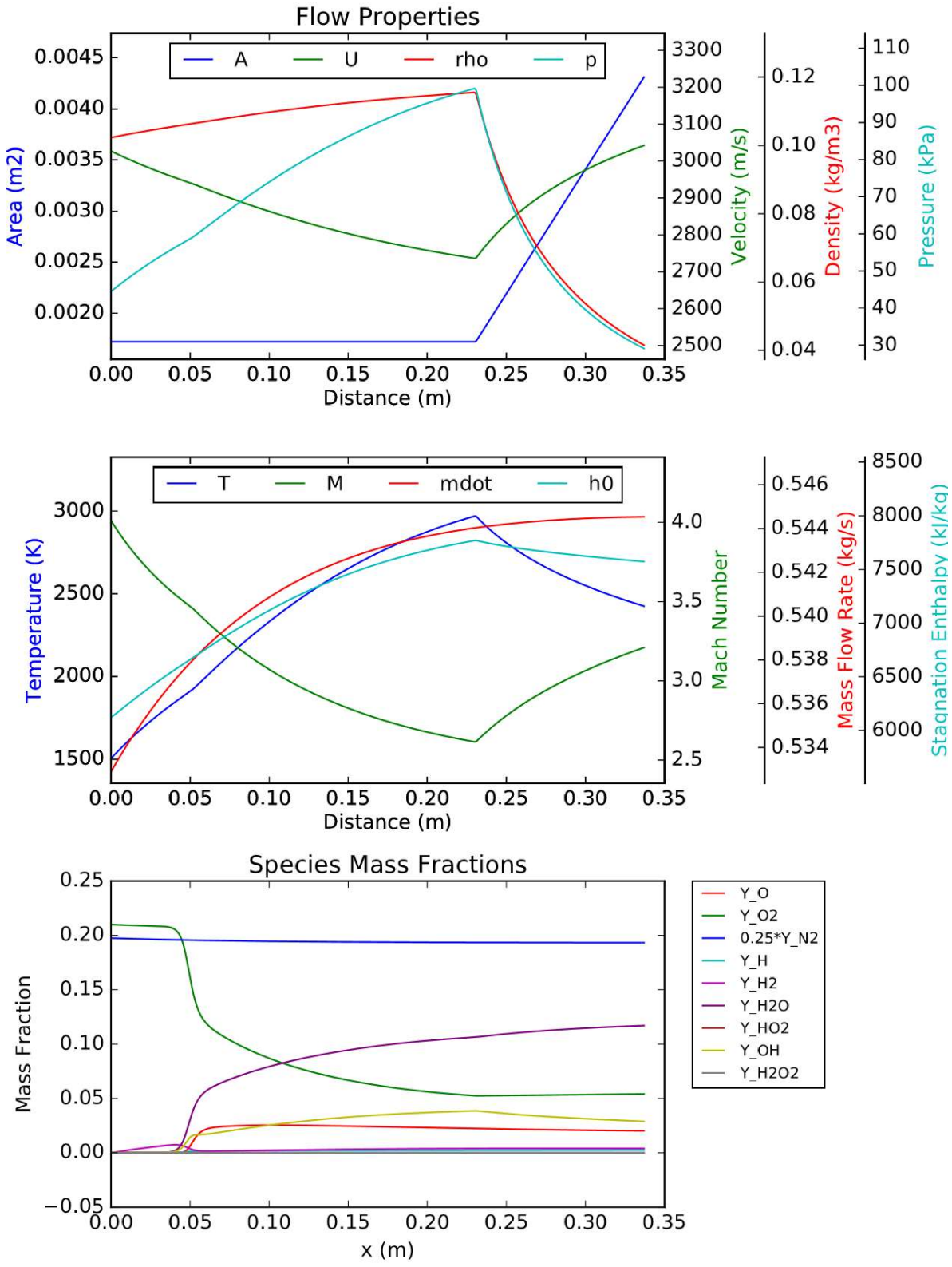

Figure 16: Mach 12 REST Engine Variation - $A_2 = 2.5A_1$

4.6 Other Engines

The results and analysis above demonstrate how the computational method can be quickly and effectively used to complete quasi-one-dimensional modelling of the combustor flow properties in the Mach 12 REST engine. The effect which varying geometry has on the flow properties has been highlighted, indicating the importance of optimising the area variation. As was previously discussed, the geometry and initial conditions of the code can be adjusted to model other scramjet combustors and provide useful results for these also.

Other combustors, however, are known to have lower initial temperatures; so low that simple cycle analysis shows that the flow will not ignite. Ignition does occur though, in the hot boundary layer. This computational method would be unable to model this as it uses the average core stream conditions. Possible further work on this topic could consider developing a real boundary layer profile model and coupling it with this core stream model by considering the heat transfer between the two. This would provide a more accurate model of the heat transfer and friction to the wall, whilst enhancing the robustness of the computation.

5.0 Conclusions

The goal of the project was to determine the optimal geometry that minimises combustor length while maximising heat release, by supporting rapid combustion initiation upstream, and minimising product dissociation downstream. The project aim, of developing a scramjet combustor model that uses computational tools, existing flow solvers and reaction mechanisms was successfully completed. Quasi-one-dimensional analysis of the scramjet combustor provided a set of Ordinary Differential Equations (ODEs) [5] [2], which were developed into a computational routine. A Python ODE solver was used to march along the scramjet length and compute flow properties from the set of ODEs. Existing thermo-chemistry routines from Eilmer [14] were also incorporated, along with a 9-species 18-reaction scheme [15], and specific heat curve-fits [12]. The tool developed allowed for adaptation of the geometry, initial conditions and mixing curve. Friction losses were accounted for, though heat losses to the walls were not.

The Mach 12 REST engine was considered for analysis, using initial conditions from [22] and geometry from [21]; the mixing efficiency curve was developed and used to describe the mass flow rate differential. The frictionless case was first compared to the case with friction losses; it was found that the inclusion of friction decreases combustion efficiency. The computational tool modelled ignition, with this noticeable on plots as the hydrogen mass fraction rapidly decreased, while that of water rapidly increased.

The constant area combustor and constantly diverging area combustor were next considered, and comparisons drawn to the Mach 12 REST engine case. The constant area combustor had the lowest combustion efficiency, while the constantly diverging combustor had a value approximately half way between that and the Mach 12 REST engine case. It was noted that the ignition occurred slightly later in this engine, due to the lower rate at which the temperature and pressure increased.

The location along the combustor at which divergence begins was then investigated; this was done by varying the lengths in the Mach 12 REST engine. It was found that if divergence commenced 15 mm further along the combustor the combustion efficiency was maximised. This alteration was made to the lengths, following which the magnitude of the divergence was investigated. It was found that the larger the ratio of the final area to the initial area, the higher the combustion efficiency, though the degree to which the combustor can reasonably diverge should be investigated further.

Further analysis should consider calculating the thrust which would be produced by each engine once the flow had been expanded in a nozzle. This would equalise the results and take further losses into account. Future work should also consider the errors encountered when attempting to include heat losses to the walls, and the development of a real boundary layer profile which could be coupled with this model to allow for modelling of boundary layer ignition cases.

Originally, the aim was to also develop an optimisation routine into which this scramjet combustor model could be embedded; time constraints resulted in this being removed from the project scope. The above results have successfully displayed how area variations can affect the flow properties, with the development of an optimisation routine into which this can be embedded recommended for future work.

6.0 References

[1] M. K. Smart, "How Much Compression Should a Scramjet Inlet Do?," *American Institure of Aeronautics and Astronautics,* pp. 610-619, 2012.

[2] C. Birzer and C. J. Doolan, "Quasi-One-Dimensional Model of Hydrogen-Fueled Scramjet Combustors," *Journal of Propulsion and Power,* pp. 1220-1225, 2009.

[3] M. K. Smart and M. R. Tetlow, "Orbital Delivery of Small Payloads Using Hypersonic Airbreathing Propulsion," *Jornal of Spacecraft and Rockets,* vol. 46, pp. 117-125, 2009.

[4] C. Fureby, M. Chapuis, E. Fedina and S. Karl, "CFD analysis of the HyShot II scramjet combustor," Elsevier, Stockholm, Sweden, 2011.

[5] T. F. O'Brien, R. P. Starkey and M. J. Lewis, "Quasi-One-Dimensional High-Speed Enginer Model with Finite-Rate Chemistry," *Journal of Propulsion and Power,* pp. 1366-1374, 2001.

[6] S. R. Turns, in *An Introduction to Combustion*, New York, McGraw-Hill , 1996, pp. 151-207.

[7] J. D. Anderson, Hypersonic and High Temperature Gas Dynamics, Reston, VA: American Institute of Aeronautics and Astronautics, 2000.

[8] G. D. Byrne, A. C. Hindmarch and P. N. Brown, "VODPK: Variable-Coefficient Ordinary Differential Equation Solver with the Preconditions Krylov Method GMRES for the Solution of Linear Systems," Lawrence Livermore National Labs, Livermore, CA, 1997.

[9] R. J. Kee, F. M. Rupley and J. A. Miller, "CHEMKIN-II: A Fortran Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics," Sandia National Labs, Albuquerque, NM, 1989.

[10] R. C. Rogers, "Mixing of Hydrogen Injected from Multiple Injectors Normal to a Supersonic Airstream," NASA, 1971.

[11] J. R. Henry and G. Y. Anderson, "Design Considerations for the Airframe-Integrated Scramjet," NASA, 1973.

[12] B. J. McBride, M. J. Zehe and S. Gordon, "NASA Glenn Goefficients for Calculating Thermodynamic Properties of Individual Species," NASA Center for Aerospace Information, Hanover, 2002.

[13] N. M. Marinov, C. K. Westbrook and W. J. Pitz, "Detailed and Global Chemical Kinetics Model for Hydrogen," Lawrence Livermore National Laboratory, Livermore, CA, 1995.

[14] Compressible-Flow CFD Group, "The CFCFD Code Collection: Eilmer," University of Queensland, Brisbane, 2016.

[15] D. A. Bittker and V. J. Scullin, "General Chemical Kinetics Computer Program for Static and Flow Reactions, with Application to Combustion and Shock-Tube Kinetics". USA Patent NASA TN D-6586, January 1972.

[16] V. Wheatley, "e3rates.py," Brisbane, 2016.

[17] The CFCFD Group, "h2-air.lua," Brisbane, 2016.

[18] F. Zander, "Bittker_Scullin.lau," The CFCFD Group, Brisbane, 2016.

[19] MathWorks, "Ordinary Differential Equations," MathWorks, 2016. [Online]. Available: https://au.mathworks.com/help/matlab/ordinary-differential-equations.html. [Accessed April 2016].

[20] SciPy.org, "SciPy v0.18.1 Reference Guide," 19 September 2016. [Online]. Available: https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.odeint.html. [Accessed 18 October 2016].

[21] J. E. Barth, D. J. Wise, V. Wheatley and M. K. Smart, "Tailored Fuel Injection for Performance Enhancement in a Mach 12 Scramjet Engine," in *20th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, Glasgow, 2015.

[22] J. Barth, "Mixing and Combustion Enhancement in a Mach 12 Shape-Transitioning Scramjet Engine," The University of Queensland, School of Mechanical and Mining Engineering, Centre for Hypersonics, Brisbane, 2014.

Appendices

Appendix A: Variables and Nomenclature

A = geometric area of the duct, m^2

$a - d, f$ = curve-fit constants

a_{1-7} = series of constants

c = speed of sound, m/s

C_f = skin-friction coefficient

C_H = Stanton number

c_p = specific heat, kJ/kg

$\mathcal{D}$ = hydraulic diameter, m

f_{st} = stoichiometric fuel/air ratio

g = gravitational acceleration, m/s^2

h = enthalpy per unit mass, J/kg

L = length, m

M = Mach number

MW = molecular weight, kg/kmol

$\overline{MW}$ = mixture molecular weight, kg/kmol

$\dot{m}$ = mass flow rate, kg/s

P_w = wetted perimeter, m

Pr = Prandtl number

p = pressure, N/m^2

$\dot{Q}$ = heat-transfer weight, J/s

R = universal gas constant, J/(kg-K)

R_U = universal gas constant, J/(kmol-K)

T = temperature, K

U = velocity, m/s

x = axial coordinator, m

$\bar{x}$ = non-dimensional axial coordinator

Y = mass fraction

γ = ratio of specific heats

Δx = control volume increment of x

ε = ratio of gas injection velocity to freestream velocity

η = efficiency

ρ = density, kg/m^3

ϕ = equivalence ratio

$\dot{\omega}$ = molar production rate, kmol/(s-m^3)

[ ] = concentration

Subscripts

$added$ = species added for fuel injection

aw = adiabatic wall

$comb$ = combustor

e = end

eff = effective

f = total fuel available

i = ith species

inj = injection

mix = missing

0 = total or stagnation conditions

r = available for reaction

ref = at reference conditions

s = start

st = stoichiometric

w = wall

$x,\ \Delta x$ = control volume descriptors

∞ = freestream conditions

Superscripts

$''$ = per unit area

$'''$ = per unit volume

$*$ = evaluated at the reference temperature

Appendix B: Specific Heat Curve Coefficients and Standard State Enthalpy Values

The following table contains the curve-fit constants and standard state enthalpy values from [12], relevant to Equation 22, for the species used in the reaction mechanism presented in [15]. In some cases, curve-fit values extended past 6000 K, up to 20000 K, however these were not included in the table as the expected temperatures in the combustor do not exceed 6000 K.

Table 4: Specific Heat Curve Coefficients and Standard State Enthalpy Values

Species	**Standard State Enthalpy (kJ/mol)**	**Temperature Range (K)**	**Curve-fit Constants**
O	242.450	200 – 1000	$a_1 = -7.953611300\times10^{03}$ $a_2 = 1.607177787\times10^{02}$ $a_3 = 1.966226438\times10^{0}$ $a_4 = 1.013670310\times10^{-03}$ $a_5 = -1.110415423\times10^{-06}$ $a_6 = 6.517507500\times10^{-10}$ $a_7 = -1.584779251\times10^{-13}$
		1000 – 6000	$a_1 = 2.619020262\times10^{05}$ $a_2 = -7.298722030\times10^{02}$ $a_3 = 3.317177270\times10^{00}$ $a_4 = -4.281334360\times10^{-04}$ $a_5 = 1.036104594\times10^{-07}$ $a_6 = -9.438304330\times10^{-12}$ $a_7 = 2.725038297\times10^{-16}$
O_2	-8.680	200 – 1000	$a_1 = -3.425563420\times10^{04}$ $a_2 = 4.847000970\times10^{02}$ $a_3 = 1.119010961\times10^{00}$ $a_4 = 4.293889240\times10^{-03}$ $a_5 = -6.836300520\times10^{-07}$ $a_6 = -2.023372700\times10^{-09}$ $a_7 = 1.039040018\times10^{-12}$
		1000 – 6000	$a_1 = -1.037939022\times10^{06}$ $a_2 = 2.344830282\times10^{03}$ $a_3 = 1.819732036e\times10^{00}$ $a_4 = 1.267847582\times10^{-0}$ $a_5 = -2.188067988\times10^{-0}$ $a_6 = 2.053719572\times10^{-11}$ $a_7 = -8.193467050\times10^{-16}$
N_2	-8.670	200 – 1000	$a_1 = 2.210371497\times10^{04}$ $a_2 = -3.818461820\times10^{02}$ $a_3 = 6.082738360\times10^{00}$ $a_4 = -8.530914410\times10^{-03}$ $a_5 = 1.384646189\times10^{-05}$ $a_6 = -9.625793620\times10^{-09}$

			$a_7 = 2.519705809\times10^{-12}$
		1000 – 6000	$a_1 = 5.877124060\times10^{05}$ $a_2 = -2.239249073\times10^{03}$ $a_3 = 6.066949220\times10^{00}$ $a_4 = -6.139685500\times10^{-04}$ $a_5 = 1.491806679\times10^{-07}$ $a_6 = -1.923105485\times10^{-11}$ $a_7 = 1.061954386\times10^{-15}$
H	211.801	200 – 1000	$a_1 = 0.000000000\times10^{00}$ $a_2 = 0.000000000\times10^{00}$ $a_3 = 2.500000000\times10^{00}$ $a_4 = 0.000000000\times10^{00}$ $a_5 = 0.000000000\times10^{00}$ $a_6 = 0.000000000\times10^{00}$ $a_7 = 0.000000000\times10^{00}$
		1000 – 6000	$a_1 = 6.078774250\times10^{01}$ $a_2 = -1.819354417\times10^{-01}$ $a_3 = 2.500211817\times10^{00}$ $a_4 = -1.226512864\times10^{-07}$ $a_5 = 3.732876330\times10^{-11}$ $a_6 = -5.687744560\times10^{-15}$ $a_7 = 3.410210197\times10^{-19}$
H_2	-8.468	200 – 1000	$a_1 = 4.078323210\times10^{04}$ $a_2 = -8.009186040\times10^{02}$ $a_3 = 8.214702010\times10^{00}$ $a_4 = -1.269714457\times10^{-02}$ $a_5 = 1.753605076\times10^{-05}$ $a_6 = -1.202860270\times10^{-08}$ $a_7 = 3.368093490\times10^{-12}$
		1000 – 6000	$a_1 = 5.608128010\times10^{05}$ $a_2 = -8.371504740\times10^{02}$ $a_3 = 2.975364532\times10^{00}$ $a_4 = 1.252249124\times10^{-03}$ $a_5 = -3.740716190\times10^{-07}$ $a_6 = 5.936625200\times10^{-11}$ $a_7 = -3.606994100\times10^{-15}$
H_2O	-251.730	200 – 1000	$a_1 = -3.947960830\times10^{04}$ $a_2 = 5.755731020\times10^{02}$ $a_3 = 9.317826530\times10^{-01}$ $a_4 = 7.222712860\times10^{-03}$ $a_5 = -7.342557370\times10^{-06}$ $a_6 = 4.955043490\times10^{-09}$ $a_7 = -1.336933246\times10^{-12}$
		1000 – 6000	$a_1 = 1.034972096\times10^{06}$ $a_2 = -2.412698562\times10^{03}$ $a_3 = 4.646110780\times10^{00}$ $a_4 = 2.291998307\times10^{-03}$ $a_5 = -6.836830480\times10^{-07}$ $a_6 = 9.426468930\times10^{-11}$ $a_7 = -4.822380530\times10^{-15}$

HO_2	2.018	200 – 1000	$a_1 = -7.598882540\times10^{04}$ $a_2 = 1.329383918\times10^{03}$ $a_3 = -4.677388240\times10^{00}$ $a_4 = 2.508308202\times10^{-02}$ $a_5 = -3.006551588\times10^{-05}$ $a_6 = 1.895600056\times10^{-08}$ $a_7 = -4.828567390\times10^{-12}$
		1000 – 6000	$a_1 = -1.810669724\times10^{06}$ $a_2 = 4.963192030\times10^{03}$ $a_3 = -1.039498992\times10^{00}$ $a_4 = 4.560148530\times10^{-03}$ $a_5 = -1.061859447\times10^{-06}$ $a_6 = 1.144567878\times10^{-10}$ $a_7 = -4.763064160\times10^{-15}$
OH	28.465	200 – 1000	$a_1 = -1.998858990\times10^{03}$ $a_2 = 9.300136160\times10^{01}$ $a_3 = 3.050854229\times10^{00}$ $a_4 = 1.529529288\times10^{-03}$ $a_5 = -3.157890998\times10^{-06}$ $a_6 = 3.315446180\times10^{-09}$ $a_7 = -1.138762683\times10^{-12}$
		1000 – 6000	$a_1 = 1.017393379\times10^{06}$ $a_2 = -2.509957276\times10^{03}$ $a_3 = 5.116547860\times10^{00}$ $a_4 = 1.305299930\times10^{-04}$ $a_5 = -8.284322260\times10^{-08}$ $a_6 = 2.006475941\times10^{-11}$ $a_7 = -1.556993656\times10^{-15}$
H_2O_2	-147.039	200 – 1000	$a_1 = -9.279533580\times10^{04}$ $a_2 = 1.564748385\times10^{03}$ $a_3 = -5.976460140\times10^{00}$ $a_4 = 3.270744520\times10^{-02}$ $a_5 = -3.932193260\times10^{-05}$ $a_6 = 2.509255235\times10^{-0}$ $a_7 = -6.465045290\times10^{-12}$
		1000 – 6000	$a_1 = 1.489428027\times10^{06}$ $a_2 = -5.170821780\times10^{03}$ $a_3 = 1.128204970\times10^{01}$ $a_4 = -8.042397790\times10^{-05}$ $a_5 = -1.818383769\times10^{-08}$ $a_6 = 6.947265590\times10^{-12}$ $a_7 = -4.827831900\times10^{-16}$

Appendix C: Combustion of Hydrogen in air - 9-Species, 18-Reaction Scheme

The 9-species, 18-reaction scheme presented in [15] includes $O, O_2, N_2, H, H_2, H_2O, OH_2, OH,$ and H_2O_2; the reactions are listed below. M is used to represent an inert molecule in reactions, which in this case corresponds to the non-reacting N_2.

$$H_2 + OH \leftrightarrow H_2O + H$$

$$H + O_2 \leftrightarrow OH + O$$

$$O + H_2 \leftrightarrow OH + H$$

$$H + O_2 + M \leftrightarrow HO_2 + M$$

$$H + H + M \leftrightarrow H_2 + M$$

$$H_2 + HO_2 \leftrightarrow H_2O_2 + H$$

$$M + H_2O_2 \leftrightarrow OH + OH + M$$

$$HO_2 + H \leftrightarrow OH + OH$$

$$H + OH + M \leftrightarrow H_2O + M$$

$$O + O + M \leftrightarrow O_2 + M$$

$$O + H_2O \leftrightarrow OH + OH$$

$$H_2 + O_2 \leftrightarrow OH + OH$$

$$HO_2 + OH \leftrightarrow H_2O + O_2$$

$$HO_2 + O \leftrightarrow O_2 + OH$$

$$HO_2 + HO_2 \leftrightarrow H_2O_2 + O_2$$

$$OH + H_2O_2 \leftrightarrow H_2O + HO_2$$

$$O + H_2O_2 \leftrightarrow OH + HO_2$$

$$H + H_2O_2 \leftrightarrow H_2O + OH$$

Appendix D: Scramjet Combustor Code

```
# Quasi-1D scramjet combustor model with finite rate chemistry
# Based on solution method presented in paper by T. O'Brien, R. Starkey and
M. Lewis (2001)
# Using odeint to solve the stiff set of ODES which characterise the flow
properties in Scramjet Combustors
# Adjusting geometry and initial conditions will enable modelling of
varying combustor designs
# Marguerite Taylor 2016
# SI units used in all cases

import sys, os, math
sys.path.append(os.path.expandvars("$HOME/e3bin"))
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.axes_grid1 import host_subplot
import mpl_toolkits.axisartist as AA
from scipy.integrate import odeint
from e3rates import e3_formation_rates

# ------------------------------------#
# ------------------------------------#
#           Specific Heats            #
# ------------------------------------#
# ------------------------------------#

def specific_heats(T,species):
    # function to calculate the specific heat of species at given
temperature
    # O, O2, N2, H, H2, H2O, HO2, OH, H2O2 - in this order for spicies
count

    if (species == 1): # O
        if (200 <= T) and (T <= 1000):
            a1 = -7.953611300e+03
            a2 = 1.607177787e+02
            a3 = 1.966226438e+00
            a4 = 1.013670310e-03
            a5 = -1.110415423e-06
            a6 = 6.517507500e-10
            a7 = -1.584779251e-13
        elif (1000 < T) and (T <= 6000):
            a1 = 2.619020262e+05
            a2 = -7.298722030e+02
            a3 = 3.317177270e+00
            a4 = -4.281334360e-04
            a5 = 1.036104594e-07
            a6 = -9.438304330e-12
            a7 = 2.725038297e-16
        elif (6000 < T) and (T <= 20000):
            a1 = 1.779004264e+08
            a2 = -1.082328257e+05
            a3 = 2.810778365e+01
            a4 = -2.975232262e-03
            a5 = 1.854997534e-07
            a6 = -5.796231540e-12
            a7 = 7.191720164e-17
        MW = MW_O

    elif (species == 2): # O2
```

```
        if (200 <= T) and (T <= 1000):
            a1 = -3.425563420e+04
            a2 = 4.847000970e+02
            a3 = 1.119010961e+00
            a4 = 4.293889240e-03
            a5 = -6.836300520e-07
            a6 = -2.023372700e-09
            a7 = 1.039040018e-12
        elif (1000 < T) and (T <= 6000):
            a1 = -1.037939022e+06
            a2 = 2.344830282e+03
            a3 = 1.819732036e+00
            a4 = 1.267847582e-03
            a5 = -2.188067988e-07
            a6 = 2.053719572e-11
            a7 = -8.193467050e-16
        elif (6000 < T) and (T <= 20000):
            a1 = 4.975294300e+08
            a2 = -2.866106874e+05
            a3 = 6.690352250e+01
            a4 = -6.169959020e-03
            a5 = 3.016396027e-07
            a6 = -7.421416600e-12
            a7 = 7.278175770e-17
        MW = MW_O2

    elif (species == 3): # N2
        if (200 <= T) and (T <= 1000):
            a1 = 2.210371497e+04
            a2 = -3.818461820e+02
            a3 = 6.082738360e+00
            a4 = -8.530914410e-03
            a5 = 1.384646189e-05
            a6 = -9.625793620e-09
            a7 = 2.519705809e-12
        elif (1000 < T) and (T <= 6000):
            a1 = 5.877124060e+05
            a2 = -2.239249073e+03
            a3 = 6.066949220e+00
            a4 = -6.139685500e-04
            a5 = 1.491806679e-07
            a6 = -1.923105485e-11
            a7 = 1.061954386e-15
        elif (6000 < T) and (T <= 20000):
            a1 = 8.310139160e+08
            a2 = -6.420733540e+05
            a3 = 2.020264635e+02
            a4 = -3.0650920463-02
            a5 = 2.486903333e-06
            a6 = -9.705954110e-11
            a7 = 1.437538881e-15
        MW = MW_N2
        # when plotted it appears that there is a problem with the values
for
        # (6000 < T < 20000) but temperature not expected to surpass this

    elif (species == 4): # H
        if (200 <= T) and (T <= 1000):
            a1 = 0.000000000e+00
            a2 = 0.000000000e+00
            a3 = 2.500000000e+00
            a4 = 0.000000000e+00
            a5 = 0.000000000e+00
```

```
            a6 = 0.000000000e+00
            a7 = 0.000000000e+00
        elif (1000 < T) and (T <= 6000):
            a1 = 6.078774250e+01
            a2 = -1.819354417e-01
            a3 = 2.500211817e+00
            a4 = -1.226512864e-07
            a5 = 3.732876330e-11
            a6 = -5.687744560e-15
            a7 = 3.410210197e-19
        elif (6000 < T) and (T <= 20000):
            a1 = 2.173757694e+08
            a2 = -1.312035403e+05
            a3 = 3.399174200e+01
            a4 = -3.813999680e-03
            a5 = 2.432854837e-07
            a6 = -7.694275540e-12
            a7 = 9.644105630e-17
        MW = MW_H

    elif (species == 5): # H2
        if (200 <= T) and (T <= 1000):
            a1 = 4.078323210e+04
            a2 = -8.009186040e+02
            a3 = 8.214702010e+00
            a4 = -1.269714457e-02
            a5 = 1.753605076e-05
            a6 = -1.202860270e-08
            a7 = 3.368093490e-12
        elif (1000 < T) and (T <=6000):
            a1 = 5.608128010e+05
            a2 = -8.371504740e+02
            a3 = 2.975364532e+00
            a4 = 1.252249124e-03
            a5 = -3.740716190e-07
            a6 = 5.936625200e-11
            a7 = -3.606994100e-15
        elif (6000 < T) and (T <= 20000):
            a1 = 4.966884120e+08
            a2 = -3.147547149e+05
            a3 = 7.984121880e+01
            a4 = -8.414789210e-03
            a5 = 4.753248350e-07
            a6 = -1.371873492e-11
            a7 = 1.605461756e-16
        MW = MW_H2

    elif (species == 6): # H2O
        if (200 <= T) and (T <= 1000):
            a1 = -3.947960830e+04
            a2 = 5.755731020e+02
            a3 = 9.317826530e-01
            a4 = 7.222712860e-03
            a5 = -7.342557370e-06
            a6 = 4.955043490e-09
            a7 = -1.336933246e-12
        elif (1000 < T) and (T <= 6000):
            a1 = 1.034972096e+06
            a2 = -2.412698562e+03
            a3 = 4.646110780e+00
            a4 = 2.291998307e-03
            a5 = -6.836830480e-07
            a6 = 9.426468930e-11
```

```
            a7 = -4.822380530e-15
        MW = MW_H2O

    elif (species == 7): # HO2
        if (200 <= T) and (T <= 1000):
            a1 = -7.598882540e+04
            a2 = 1.329383918e+03
            a3 = -4.677388240e+00
            a4 = 2.508308202e-02
            a5 = -3.006551588e-05
            a6 = 1.895600056e-08
            a7 = -4.828567390e-12
        elif (1000 < T) and (T <= 6000):
            a1 = -1.810669724e+06
            a2 = 4.963192030e+03
            a3 = -1.039498992e+00
            a4 = 4.560148530e-03
            a5 = -1.061859447e-06
            a6 = 1.144567878e-10
            a7 = -4.763064160e-15
        MW = MW_HO2

    elif (species == 8): # OH
        if (200 <= T) and (T <= 1000):
            a1 = -1.998858990e+03
            a2 = 9.300136160e+01
            a3 = 3.050854229e+00
            a4 = 1.529529288e-03
            a5 = -3.157890998e-06
            a6 = 3.315446180e-09
            a7 = -1.138762683e-12
        elif (1000 < T) and (T <= 6000):
            a1 = 1.017393379e+06
            a2 = -2.509957276e+03
            a3 = 5.116547860e+00
            a4 = 1.305299930e-04
            a5 = -8.284322260e-08
            a6 = 2.006475941e-11
            a7 = -1.556993656e-15
        elif (6000 < T) and (T <= 20000):
            a1 = 2.847234193e+08
            a2 = -1.859532612e+05
            a3 = 5.008240900e+01
            a4 = -5.142374980e-03
            a5 = 2.875536589e-07
            a6 = -8.228817960e-12
            a7 = 9.567229020e-17
        MW = MW_OH

    elif (species == 9): # H2O2
        if (200 <= T) and (T <= 1000):
            a1 = -9.279533580e+04
            a2 = 1.564748385e+03
            a3 = -5.976460140e+00
            a4 = 3.270744520e-02
            a5 = -3.932193260e-05
            a6 = 2.509255235e-08
            a7 = -6.465045290e-12
        elif (1000 < T) and (T <= 6000):
            a1 = 1.489428027e+06
            a2 = -5.170821780e+03
            a3 = 1.128204970e+01
            a4 = -8.042397790e-05
```

```
            a5 = -1.818383769e-08
            a6 = 6.947265590e-12
            a7 = -4.827831900e-16
        MW = MW_H2O2

    Cp = (R/MW)*(a1*T**-2 + a2*T**-1 + a3 + a4*T + a5*T**2 + a6*T**3 +
a7*T**4)
    Cps = (R/MW)*(-2*a1*T**-3 - a2*T**-2 + a4 + 2*a5*T + 3*a6*T**2 +
4*a7*T**3)
    return Cp, Cps

# ------------------------------------#
# ------------------------------------#
#               ODE's                 #
# ------------------------------------#
# ------------------------------------#
def diff(y,x):

    U = y[0]
    rho = y[1]
    p = y[2]
    Y_O = y[3]
    Y_O2 = y[4]
    Y_N2 = y[5]
    Y_H = y[6]
    Y_H2 = y[7]
    Y_H2O = y[8]
    Y_HO2 = y[9]
    Y_OH = y[10]
    Y_H2O2 = y[11]

    sumY = Y_O + Y_O2 + Y_N2 + Y_H + Y_H2 + Y_H2O + Y_HO2 + Y_OH + Y_H2O2
    #if sumY != 1:
        #print "The species mass fractions sum to", sumY

    # ----------------------------------
    #           Flow Properties
    # ----------------------------------
    T = p/(rho*Rsp)
    #print 'Diff', x, y, T

    [Cp_O,Cps_O] = specific_heats(T,1)
    [Cp_O2,Cps_O2] = specific_heats(T,2)
    [Cp_N2,Cps_N2] = specific_heats(T,3)
    [Cp_H,Cps_H] = specific_heats(T,4)
    [Cp_H2,Cps_H2] = specific_heats(T,5)
    [Cp_H2O,Cps_H2O] = specific_heats(T,6)
    [Cp_HO2,Cps_HO2] = specific_heats(T,7)
    [Cp_OH,Cps_OH] = specific_heats(T,8)
    [Cp_H2O2,Cps_H2O2] = specific_heats(T,9)

    Cv_O = Cp_O - R/MW_O
    Cv_O2 = Cp_O2 - R/MW_O2
    Cv_N2 = Cp_N2 - R/MW_N2
    Cv_H = Cp_H - R/MW_H
    Cv_H2 = Cp_H2 - R/MW_H2
    Cv_H2O = Cp_H2O - R/MW_H2O
    Cv_HO2 = Cp_HO2 - R/MW_HO2
    Cv_OH = Cp_OH - R/MW_OH
    Cv_H2O2 = Cp_H2O2 - R/MW_H2O2
```

```
    Cp = Cp_O*Y_O + Cp_O2*Y_O2 + Cp_N2*Y_N2 + Cp_H*Y_H + Cp_H2*Y_H2 +
Cp_H2O*Y_H2O + Cp_HO2*Y_HO2 + Cp_OH*Y_OH + Cp_H2O2*Y_H2O2
    Cv = Cv_O*Y_O + Cv_O2*Y_O2 + Cv_N2*Y_N2 + Cv_H*Y_H + Cv_H2*Y_H2 +
Cv_H2O*Y_H2O + Cv_HO2*Y_HO2 + Cv_OH*Y_OH + Cv_H2O2*Y_H2O2
    gamma = Cp/Cv
    MW = 1/(Y_O/MW_O + Y_O2/MW_O2 + Y_N2/MW_N2 + Y_H/MW_H + Y_H2/MW_H2 +
Y_H2O/MW_H2O + Y_HO2/MW_HO2 + Y_OH/MW_OH + Y_H2O2/MW_H2O2)
    Cps = Cp + T*(Cps_O*Y_O + Cps_O2*Y_O2 + Cps_N2*Y_N2 + Cps_H*Y_H +
Cps_H2*Y_H2 + Cps_H2O*Y_H2O + Cps_HO2*Y_HO2 + Cps_OH*Y_OH +
Cps_H2O2*Y_H2O2)

    M = U/((gamma*p/rho)**0.5)
    Taw = Tw #T*(1 + (Pr**(1/3))*((gamma-1)/2)*M**2) # set to Tw for no
heat loss at walls
    h_O = hf_O + Cp_O*(T-Tref)
    h_O2 = hf_O2 + Cp_O2*(T-Tref)
    h_N2 = hf_N2 + Cp_N2*(T-Tref)
    h_H = hf_H + Cp_H*(T-Tref)
    h_H2 = hf_H2 + Cp_H2*(T-Tref)
    h_H2O = hf_H2O + Cp_H2O*(T-Tref)
    h_HO2 = hf_HO2 + Cp_HO2*(T-Tref)
    h_OH = hf_OH + Cp_OH*(T-Tref)
    h_H2O2 = hf_H2O2 + Cp_H2O2*(T-Tref)

    # ------------------------------------
    #           Area profile
    # ------------------------------------
    if x <= L1:
        dAondx = 0
        A = A1
    else:
        dAondx = (A2-A1)/(L2)
        A = A1+dAondx*(x-L1)
    Dh = (A*(4/math.pi))**0.5 # hydraulic diameter

    # ------------------------------------
    #        Mass mixing profile
    # ------------------------------------
    # if phi = 0, mdotf = 0, so dmdotondx = 0;
    mdot = rho*U*A
    #dmdotondx = 0
    #dmdotondx = mdotf/L
    dmdotondx = mdotf*(a*b*math.exp(b*x) + c*d*math.exp(d*x)) # based on
curve-fit to entrainment efficiency

    # ------------------------------------
    #   Species Mass Fraction Profiles
    # ------------------------------------
    [w_O,w_O2,w_N2,w_H,w_H2,w_H2O,w_HO2,w_OH,w_H2O2] =
e3_formation_rates(T,p,Y_O,Y_O2,Y_N2,Y_H,Y_H2,Y_H2O,Y_HO2,Y_OH,Y_H2O2) #
Reaction rates # 0 if no reactions

    # change in species mass fractions
    dYondx_O = (w_O*MW_O)/(rho*U) - Y_O*dmdotondx/mdot
    dYondx_O2 = (w_O2*MW_O2)/(rho*U) - Y_O2*dmdotondx/mdot
    dYondx_N2 = (w_N2*MW_N2)/(rho*U) - Y_N2*dmdotondx/mdot
    dYondx_H = (w_H*MW_H)/(rho*U) - Y_H*dmdotondx/mdot
    dYondx_H2 = (w_H2*MW_H2)/(rho*U) - Y_H2*dmdotondx/mdot + dmdotondx/mdot
# extra term for added fuel
    dYondx_H2O = (w_H2O*MW_H2O)/(rho*U) - Y_H2O*dmdotondx/mdot
    dYondx_HO2 = (w_HO2*MW_HO2)/(rho*U) - Y_HO2*dmdotondx/mdot
    dYondx_OH = (w_OH*MW_OH)/(rho*U) - Y_OH*dmdotondx/mdot
```

```
    dYondx_H2O2 = (w_H2O2*MW_H2O2)/(rho*U) - Y_H2O2*dmdotondx/mdot

    # ------------------------------------
    #  Mixture molecular weight profile
    # ------------------------------------
    term1 = dYondx_O/MW_O + dYondx_O2/MW_O2 + dYondx_N2/MW_N2 +
dYondx_H/MW_H + dYondx_H2/MW_H2 + dYondx_H2O/MW_H2O + dYondx_HO2/MW_HO2 +
dYondx_OH/MW_OH + dYondx_H2O2/MW_H2O2
    dMWondx = (-MW**2)*term1

    # ------------------------------------
    #             Enthalpies
    # ------------------------------------
    Cp_hat = Cps - (mdotf/mdot)*(Cp_H2 + Cps_H2*T)
    h_hat = Cp_hat*T
    h0 = (U**2)/2 + h_O*Y_O + h_O2*Y_O2 + h_N2*Y_N2 + h_H*Y_H + h_H2*Y_H2 +
h_H2O*Y_H2O + h_HO2*Y_HO2 + h_OH*Y_OH + h_H2O2*Y_H2O2

    # ------------------------------------
    #          Velocity profile
    # ------------------------------------
    alpha = (1/U)*(1 - gamma*M**2 + (U**2)/h_hat)
    term2 = (-1/A)*dAondx
    term3 = ((1+gamma*M**2*(1-epsilon)-(h0/h_hat))/mdot)*dmdotondx # should
be zero if no added fuel
    term4 = h_O*dYondx_O + h_O2*dYondx_O2 + h_N2*dYondx_N2 + h_H*dYondx_H +
h_H2*dYondx_H2 + h_H2O*dYondx_H2O + h_HO2*dYondx_HO2 + h_OH*dYondx_OH +
h_H2O2*dYondx_H2O2 # should be 0 if no mixing
    term5 = (1/mdot)*(h_H2*dmdotondx) # should be zero if no added fuel
    term6 = (1/h_hat)*(-term4+term5)
    term7 = (-1/MW)*dMWondx
    term8 = (gamma*M**2 - (Cp*(Taw-Tw))/(h_hat*A*Pr**(2/3)))*(2*Cf/Dh) #
heat loss term, 0 if Taw = Tw, Cf = 0
    dUondx = (1/alpha)*(term2+term3+term6+term7+term8)

    # ------------------------------------
    #          Density Profile
    # ------------------------------------
    drhoondx = rho*(dmdotondx/mdot - dUondx/U - dAondx/A); # first term 0
if no addded fuel

    # ------------------------------------
    #             Pressure
    # ------------------------------------
    dpondx = -rho*U**2 * ((dUondx/U) + 2*Cf/Dh + ((1-
epsilon)/mdot)*dmdotondx); # last term 0 if no added fuel, second last term
0 if no fricition

    # Output derivatives for ODE solver
    return [dUondx, drhoondx, dpondx, dYondx_O, dYondx_O2, dYondx_N2,
dYondx_H, dYondx_H2, dYondx_H2O, dYondx_HO2, dYondx_OH, dYondx_H2O2]

# ------------------------------------#
# ------------------------------------#
#             COMBUSTOR              #
# ------------------------------------#
# ------------------------------------#

# global parameters
global Pr, Tw, f, phi, Linj, A1, A2, L1, L2, epsilon, a, b, c, d, g
global Cf, Tref, mdot0, mdotf, R, Rsp
global MW_O, MW_O2, MW_N2, MW_H, MW_H2, MW_H2O, MW_HO2, MW_OH, MW_H2O2
```

```
global hf_O, hf_O2, hf_N2, hf_H, hf_H2, hf_H2O, hf_HO2, hf_OH, hf_H2O2

# ----------------------------------
#  CONSTANTS AND INITIAL CONDITIONS
# ----------------------------------
# molecular weights (kg/mol)
MW_O = 16.00e-3
MW_O2 = 32.00e-3
MW_N2 = 28.02e-3
MW_H = 1.01e-3
MW_H2 = 2.02e-3
MW_H2O = 18.02e-3
MW_HO2 = 18.02e-3
MW_OH = 17.01e-3
MW_H2O2 = 34.02e-3

R = 8.314510 # Universal gas constant (J/mol/K)

# combustor geometry
ae = 31.05e-3 # semi-major axis of ellipse
be = ae/1.76 # semi-minor axis (from aspect ratio of 1.76)
A1 = ae*be*math.pi # initial cross-sectional area of the engine (m2)
A2 = 2*A1 # final cross-section area of the engine (m2)
cfnd = 0 # difference in location of L1 to L2 transition from normal design
(mm)
L1 = (215.67-cfnd)*1e-3 # length of constant area section (m)
L2 = (121+cfnd)*1e-3 # length of expansion section (m)
L = L1 + L2 # total length

# initial conditions
A = A1
U = 3027 # velocity at x=0
p = 44.502e3 # pressure at x=0
T = 1560 #872 # temperature at x=0
Tw = 500 # wall temperature
M = 4.02 # Mach number at x=0
rho = 0.1023 # density at x=0
print 'A1', A1, 'A2', A2, 'L1', L1, 'L2', L2, 'L', L

# combustor parameters
Pr = 0.71 # Prandtl number
f = 0.0291 # stoichiometric H2
phi = 0.75 # equivalence ratio - set to 0 for no fuel addition

# Specific Heats (J/mol/K)/(kg/mol) = J/kg/K
[Cp_O,Cps_O] = specific_heats(T,1)
[Cp_O2,Cps_O2] = specific_heats(T,2)
[Cp_N2,Cp2_N2] = specific_heats(T,3)
[Cp_H,Cps_H] = specific_heats(T,4)
[Cp_H2,Cps_H2] = specific_heats(T,5)
[Cp_H2O,Cps_H2O] = specific_heats(T,6)
[Cp_HO2,Cps_HO2] = specific_heats(T,7)
[Cp_OH,Cps_OH] = specific_heats(T,8)
[Cp_H2O2,Cps_H2O2] = specific_heats(T,9)

Cv_O = Cp_O - R/MW_O
Cv_O2 = Cp_O2 - R/MW_O2
Cv_N2 = Cp_N2 - R/MW_N2
Cv_H = Cp_H - R/MW_H
Cv_H2 = Cp_H2 - R/MW_H2
Cv_H2O = Cp_H2O - R/MW_H2O
Cv_HO2 = Cp_HO2 - R/MW_HO2
```

```
Cv_OH = Cp_OH - R/MW_OH
Cv_H2O2 = Cp_H2O2 - R/MW_H2O2

# Initial mass fractions and specific heats etc
Y_O = 0
Y_O2 = 0.21
Y_N2 = 0.79
Y_H = 0
Y_H2 = 0
Y_H2O = 0
Y_HO2 = 0
Y_OH = 0
Y_H2O2 = 0
MW = 1/(Y_O/MW_O + Y_O2/MW_O2 + Y_N2/MW_N2 + Y_H/MW_H + Y_H2/MW_H2 +
Y_H2O/MW_H2O + Y_HO2/MW_HO2 + Y_OH/MW_OH + Y_H2O2/MW_H2O2)
Cp = Cp_O*Y_O + Cp_O2*Y_O2 + Cp_N2*Y_N2 + Cp_H*Y_H + Cp_H2*Y_H2 +
Cp_H2O*Y_H2O + Cp_HO2*Y_HO2 + Cp_OH*Y_OH + Cp_H2O2*Y_H2O2
Cv = Cv_O*Y_O + Cv_O2*Y_O2 + Cv_N2*Y_N2 + Cv_H*Y_H + Cv_H2*Y_H2 +
Cv_H2O*Y_H2O + Cv_HO2*Y_HO2 + Cv_OH*Y_OH + Cv_H2O2*Y_H2O2
gamma = Cp/Cv
Rsp = Cp*(1-1/gamma) # Universal Gas Constant (J/kg/K) (R/MW) - assumed
constant for simplicity

# calculated initial conditions
mdot0 = rho*U*A # initial mass flow rate
mdotf = mdot0*phi*f # mass flow rate of fuel injection

# ------------------------------------
#           Mixing Profile
# ------------------------------------
epsilon = 0 # ratio of velocity of gas injection over velocity of flowfield
a = 1.168 # curvefit constants a-d for mass flow rate
b = -0.3605
c = -1.182
d = -9.525
Cf = 2e-3 # Friction Coefficient

# ------------------------------------
#             Enthalipes
# ------------------------------------
Tref = 0 # reference temperature (0 K)
hf_O = (242.450e3)/MW_O  # enthalpies at 0 K (J/mol)/(kg/mol) = J/kg
hf_O2 = -(8.680e3)/MW_O2
hf_N2 = -(8.670e3)/MW_N2
hf_H = (211.801e3)/MW_H
hf_H2 = -(8.468e3)/MW_H2
hf_H2O = -(251.730e3)/MW_H2O
hf_HO2 = (2.018e3)/MW_HO2
hf_OH = (28.465e3)/MW_OH
hf_H2O2 = -(147.039e3)/MW_H2O2

# ------------------------------------
#             SOLVER
# ------------------------------------
dx = 0.001
INITIAL = [U,rho,p,Y_O,Y_O2,Y_N2,Y_H,Y_H2,Y_H2O,Y_HO2,Y_OH,Y_H2O2]
print 'initial :', INITIAL
XSPAN = np.arange(0,L+dx,dx)
YOUT = odeint(diff, INITIAL, XSPAN)

# ------------------------------------
#             RESULTS
```

```
# ----------------------------------
# Using results - plotting
# uses the ode outputs to determine other parameters at each step along the
combustor
size = len(XSPAN)
M_x = []
M_A = []
M_U = []
M_rho = []
M_p = []
M_T = []
M_M = []
M_mdot = []
M_T0 = []
M_h0 = []
M_gamma= []
M_YO = []
M_YO2 = []
M_YN2 = []
M_YH = []
M_YH2 = []
M_YH2O = []
M_YHO2 = []
M_YOH = []
M_YH2O2 = []
M_MW = []
for j in range(0,size):
    x = XSPAN[j]
    U = YOUT[j][0]
    rho = YOUT[j][1]
    p = YOUT[j][2]
    Y_O = YOUT[j][3]
    Y_O2 = YOUT[j][4]
    Y_N2 = YOUT[j][5]
    Y_H = YOUT[j][6]
    Y_H2 = YOUT[j][7]
    Y_H2O = YOUT[j][8]
    Y_HO2 = YOUT[j][9]
    Y_OH = YOUT[j][10]
    Y_H2O2 = YOUT[j][11]

    T = p/(rho*Rsp)

    [Cp_O,Cps_O] = specific_heats(T,1)
    [Cp_O2,Cps_O2] = specific_heats(T,2)
    [Cp_N2,Cp2_N2] = specific_heats(T,3)
    [Cp_H,Cps_H] = specific_heats(T,4)
    [Cp_H2,Cps_H2] = specific_heats(T,5)
    [Cp_H2O,Cps_H2O] = specific_heats(T,6)
    [Cp_HO2,Cps_HO2] = specific_heats(T,7)
    [Cp_OH,Cps_OH] = specific_heats(T,8)
    [Cp_H2O2,Cps_H2O2] = specific_heats(T,9)

    Cv_O = Cp_O - R/MW_O
    Cv_O2 = Cp_O2 - R/MW_O2
    Cv_N2 = Cp_N2 - R/MW_N2
    Cv_H = Cp_H - R/MW_H
    Cv_H2 = Cp_H2 - R/MW_H2
    Cv_H2O = Cp_H2O - R/MW_H2O
    Cv_HO2 = Cp_HO2 - R/MW_HO2
    Cv_OH = Cp_OH - R/MW_OH
    Cv_H2O2 = Cp_H2O2 - R/MW_H2O2
```

```
    Cp = Cp_O*Y_O + Cp_O2*Y_O2 + Cp_N2*Y_N2 + Cp_H*Y_H + Cp_H2*Y_H2 +
Cp_H2O*Y_H2O + Cp_HO2*Y_HO2 + Cp_OH*Y_OH + Cp_H2O2*Y_H2O2
    Cv = Cv_O*Y_O + Cv_O2*Y_O2 + Cv_N2*Y_N2 + Cv_H*Y_H + Cv_H2*Y_H2 +
Cv_H2O*Y_H2O + Cv_HO2*Y_HO2 + Cv_OH*Y_OH + Cv_H2O2*Y_H2O2
    gamma = Cp/Cv
    MW = 1/(Y_O/MW_O + Y_O2/MW_O2 + Y_N2/MW_N2 + Y_H/MW_H + Y_H2/MW_H2 +
Y_H2O/MW_H2O + Y_HO2/MW_HO2 + Y_OH/MW_OH + Y_H2O2/MW_H2O2)

    if x <= L1:
        A = A1
    else:
        A = A1+((A2-A1)/(L2))*(x-L1)
    Dh = (A*(4/math.pi))**0.5

    M = U/(gamma*p/rho)**0.5
    mdot = rho*U*A
    h_O = hf_O + Cp_O*(T-Tref)
    h_O2 = hf_O2 + Cp_O2*(T-Tref)
    h_N2 = hf_N2 + Cp_N2*(T-Tref)
    h_H = hf_H + Cp_H*(T-Tref)
    h_H2 = hf_H2 + Cp_H2*(T-Tref)
    h_H2O = hf_H2O + Cp_H2O*(T-Tref)
    h_HO2 = hf_HO2 + Cp_HO2*(T-Tref)
    h_OH = hf_OH + Cp_OH*(T-Tref)
    h_H2O2 = hf_H2O2 + Cp_H2O2*(T-Tref)
    h0 = (U**2)/2 + h_O*Y_O + h_O2*Y_O2 + h_N2*Y_N2 + h_H*Y_H + h_H2*Y_H2 +
h_H2O*Y_H2O + h_HO2*Y_HO2 + h_OH*Y_OH + h_H2O2*Y_H2O2
    T0 = h0/Cp

    M_x.append(x)
    M_A.append(A)
    M_U.append(U)
    M_rho.append(rho)
    M_p.append(p/1e3)
    M_T.append(T)
    M_M.append(M)
    M_mdot.append(mdot)
    M_T0.append(T0)
    M_h0.append(h0/1e3)
    M_gamma.append(gamma)
    M_YO.append(Y_O)
    M_YO2.append(Y_O2)
    M_YN2.append(0.25*Y_N2)
    M_YH.append(Y_H)
    M_YH2.append(Y_H2)
    M_YH2O.append(Y_H2O)
    M_YHO2.append(Y_HO2)
    M_YOH.append(Y_OH)
    M_YH2O2.append(Y_H2O2)
    M_MW.append(MW)
print 'final results'
print 'x', x
print 'A', A
print 'U', U
print 'rho', rho
print 'p', p
print 'T', T
print 'M', M
print 'mdot', mdot
print 'Y_O', Y_O
print 'Y_O2', Y_O2
```

```
print 'Y_N2', Y_N2
print 'Y_H', Y_H
print 'Y_H2', Y_H2
print 'Y_H2O', Y_H2O
print 'Y_HO2', Y_HO2
print 'Y_OH', Y_OH
print 'Y_H2O2', Y_H2O2

# ----------------------------------
#             PLOTTING
# ----------------------------------

# ------------TOP PLOT---------------
host = host_subplot(311, axes_class=AA.Axes)
plt.subplots_adjust(right=0.7)
plt.subplots_adjust(top=0.95)
plt.subplots_adjust(bottom=0.05)
plt.subplots_adjust(hspace=0.3)

par1 =host.twinx()
par2 = host.twinx()
par3 = host.twinx()

offset = 60
new_fixed_axis = par2.get_grid_helper().new_fixed_axis
par2.axis["right"] = new_fixed_axis(loc="right", axes=par2,
offset=(offset,0))
par3.axis["right"] = new_fixed_axis(loc="right", axes=par3,
offset=(offset*2,0))

par2.axis["right"].toggle(all=True)
par3.axis["right"].toggle(all=True)

host.set_xlim(0,0.35)
host.set_ylim(0.9*min(M_A),1.1*max(M_A))

host.set_title("Flow Properties")
host.set_xlabel("Distance (m)")
host.set_ylabel("Area (m2)")
par1.set_ylabel("Velocity (m/s)")
par2.set_ylabel("Density (kg/m3)")
par3.set_ylabel("Pressure (kPa)")

p1, = host.plot(M_x,M_A, label="A")
p2, = par1.plot(M_x,M_U, label="U")
p3, = par2.plot(M_x,M_rho, label="rho")
p4, = par3.plot(M_x,M_p, label="p")

par1.set_ylim(0.9*min(M_U),1.1*max(M_U))
par2.set_ylim(0.9*min(M_rho),1.15*max(M_rho))
par3.set_ylim(0.9*min(M_p),1.15*max(M_p))

host.legend(loc=9, ncol=4, fontsize="medium")

host.axis["left"].label.set_color(p1.get_color())
par1.axis["right"].label.set_color(p2.get_color())
par2.axis["right"].label.set_color(p3.get_color())
par3.axis["right"].label.set_color(p4.get_color())

# ------------MID PLOT---------------
host = host_subplot(312, axes_class=AA.Axes)
```

```
plt.subplots_adjust(right=0.7)

par5 = host.twinx()
par6 = host.twinx()
par7 = host.twinx()

offset = 60
new_fixed_axis = par6.get_grid_helper().new_fixed_axis
par6.axis["right"] = new_fixed_axis(loc="right", axes=par6,
offset=(offset,0))
par7.axis["right"] = new_fixed_axis(loc="right", axes=par7,
offset=(offset*2,0))

par6.axis["right"].toggle(all=True)
par7.axis["right"].toggle(all=True)

host.set_xlim(0,0.35)
host.set_ylim(0.9*min(M_T),1.12*max(M_T))

host.set_xlabel("Distance (m)")
host.set_ylabel("Temperature (K)")
par5.set_ylabel("Mach Number")
par6.set_ylabel("Mass Flow Rate (kg/s)")
par7.set_ylabel("Stagnation Enthalpy (kJ/kg)")

p5, = host.plot(M_x,M_T, label="T")
p6, = par5.plot(M_x,M_M, label="M")
p7, = par6.plot(M_x,M_mdot, label="mdot")
p8, = par7.plot(M_x,M_h0, label="h0")

par5.set_ylim(0.9*min(M_M),1.1*max(M_M))
par6.set_ylim(0.999*min(M_mdot),1.005*max(M_mdot))
par7.set_ylim(0.9*min(M_h0),1.1*max(M_h0))

host.legend(loc=9, ncol=4, fontsize="medium")

host.axis["left"].label.set_color(p5.get_color())
par5.axis["right"].label.set_color(p6.get_color())
par6.axis["right"].label.set_color(p7.get_color())
par7.axis["right"].label.set_color(p8.get_color())

# ------------LAST PLOT--------------
plt.subplot(313)
species =
['Y_O','Y_O2','0.25*Y_N2','Y_H','Y_H2','Y_H2O','Y_HO2','Y_OH','Y_H2O2']
plt.plot(M_x,M_YO,label=species[0],color='r')
plt.plot(M_x,M_YO2,label=species[1],color='g')
plt.plot(M_x,M_YN2,label=species[2],color='b')
plt.plot(M_x,M_YH,label=species[3],color='c')
plt.plot(M_x,M_YH2,label=species[4],color='m')
plt.plot(M_x,M_YH2O,label=species[5],color='purple')
plt.plot(M_x,M_YHO2,label=species[6],color='firebrick')
plt.plot(M_x,M_YOH,label=species[7],color='y')
plt.plot(M_x,M_YH2O2,label=species[8],color='gray')
plt.title('Species Mass Fractions')
plt.xlabel('x (m)')
plt.ylabel('Mass Fraction')
plt.legend(bbox_to_anchor=(1.05,1), loc=2, borderaxespad=0.,
fontsize="medium", prop = {'size':9})
plt.draw()
plt.show()
```

Appendix E: Reaction Rates Code

```
# e3rates.py
# Utilize the cfcfd3 gas and chemistry models to compute formation
# rates for species.
# Based on reacting-pipe-flow.py, PJ 21-Jun-2011
#
# VW 28/09/2016
#
# This is set up to use the Bittker-Scullin model for hydrogen combustion

#-------------------------------------------------------------

import sys, os, math
import numpy as np
sys.path.append(os.path.expandvars("$HOME/e3bin"))
from libprep3 import *

def e3_formation_rates(T, p, YO, YO2, YN2, YH, YH2, YH2O, YHO2, YOH, YH2O2)
:
    species = ['O', 'O2', 'N2', 'H', 'H2', 'H2O', 'HO2', 'OH', 'H2O2']
    #create_gas_file('thermally perfect gas', species, 'h2-air.lua')
    gmodel = create_gas_model('h2-air.lua')
    nsp = gmodel.get_number_of_species()
    gas0 = Gas_data(gmodel)
    gas0.p = p # Pa
    gas0.T[0] = T # degree K
    gas0.massf[0] = YO
    gas0.massf[1] = YO2
    gas0.massf[2] = YN2
    gas0.massf[3] = YH
    gas0.massf[4] = YH2
    gas0.massf[5] = YH2O
    gas0.massf[6] = YHO2
    gas0.massf[7] = YOH
    gas0.massf[8] = YH2O2
    gmodel.eval_thermo_state_pT(gas0)
    rupdate = create_Reaction_update('Bittker_Scullin.lua', gmodel)
    w = rupdate.rate_of_change_py(gas0)
    return w[0], w[1], w[2], w[3], w[4], w[5], w[6], w[7], w[8]
```

Appendix F: Gas Model Code

```
-- Auto-generated by gasfile on: 28-Sep-2016 13:30:48
model = 'composite gas'
equation_of_state = 'perfect gas'
thermal_behaviour = 'thermally perfect'
mixing_rule = 'Wilke'
sound_speed = 'equilibrium'
diffusion_coefficients = 'hard sphere'
ignore_mole_fraction = 1.0e-15
T_COLD = 20.0
species = {'O', 'O2', 'N2', 'H', 'H2', 'H2O', 'HO2', 'OH', 'H2O2', }

O = {}
O.atomic_constituents = {
O=1,
}
O.charge = 0
O.M = {
  value = 0.0159994,
  reference = "CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
O.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
  units = "m",
}
O.viscosity = {
  parameters = {
    {
      A = 0.77269241,
      C = -58502.098,
      B = 83.842977,
      T_high = 5000,
      T_low = 1000,
      D = 0.85100827,
    },
    {
      A = 0.87669586,
      C = -1088456.6,
      B = 1015.842,
      T_high = 15000,
      T_low = 5000,
      D = -0.18001077,
    },
    ref = "from CEA2::trans.inp which cites Levin et al (1990)",
  },
  model = "CEA",
}
O.thermal_conductivity = {
  parameters = {
    {
      A = 0.77271664,
      C = -58580.966,
      B = 83.9891,
      T_high = 5000,
      T_low = 1000,
      D = 1.51799,
```

```
    },
    {
      A = 0.87676666,
      C = -1090669,
      B = 1017.0744,
      T_high = 15000,
      T_low = 5000,
      D = 0.48644232,
    },
    ref = "from CEA2::trans.inp which cites Levin et al (1990)",
  },
  model = "CEA",
}
O.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      -7953.6113,
      160.7177787,
      1.966226438,
      0.00101367031,
      -1.110415423e-06,
      6.5175075e-10,
      -1.584779251e-13,
      28403.62437,
      8.40424182,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      261902.0262,
      -729.872203,
      3.31717727,
      -0.000428133436,
      1.036104594e-07,
      -9.43830433e-12,
      2.725038297e-16,
      33924.2806,
      -0.667958535,
    },
  },
  {
    T_high = 20000,
    T_low = 6000,
    coeffs = {
      177900426.4,
      -108232.8257,
      28.10778365,
      -0.002975232262,
      1.854997534e-07,
      -5.79623154e-12,
      7.191720164e-17,
      889094.263,
      -218.1728151,
    },
  },
  ref = "from CEA2::thermo.inp",
}
O2 = {}
O2.atomic_constituents = {
```

```
O=2,
}
O2.charge = 0
O2.M = {
  value = 0.0319988,
  reference = "from CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
O2.d = {
  value = 3.433e-10,
  reference = "Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard sphere diameter, based on L-J parameters",
  units = "m",
}
O2.viscosity = {
  ref = "from CEA2::trans.inp which cites Boushehri et al (1987) and Svehla
(1994)",
  parameters = {
    {
      A = 0.6091618,
      C = -599.74009,
      B = -52.244847,
      T_high = 1000,
      T_low = 200,
      D = 2.0410801,
    },
    {
      A = 0.72216486,
      C = -57974.816,
      B = 175.50839,
      T_high = 5000,
      T_low = 1000,
      D = 1.0901044,
    },
    {
      A = 0.73981127,
      C = -378331.68,
      B = 391.94906,
      T_high = 15000,
      T_low = 5000,
      D = 0.9093178,
    },
  },
  model = "CEA",
}
O2.thermal_conductivity = {
  ref = "from CEA2::trans.inp which cites Boushehri et al (1987) and Svehla
(1994)",
  parameters = {
    {
      A = 0.77229167,
      C = -5893.3377,
      B = 6.846321,
      T_high = 1000,
      T_low = 200,
      D = 1.2210365,
    },
    {
      A = 0.90917351,
      C = -79650.171,
      B = 291.24182,
      T_high = 5000,
```

```
      T_low = 1000,
      D = 0.064851631,
    },
    {
      A = -1.1218262,
      C = 23295011,
      B = -19286.378,
      T_high = 15000,
      T_low = 5000,
      D = 20.342043,
    },
  },
  model = "CEA",
}
O2.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      -34255.6342,
      484.700097,
      1.119010961,
      0.00429388924,
      -6.83630052e-07,
      -2.0233727e-09,
      1.039040018e-12,
      -3391.45487,
      18.4969947,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      -1037939.022,
      2344.830282,
      1.819732036,
      0.001267847582,
      -2.188067988e-07,
      2.053719572e-11,
      -8.19346705e-16,
      -16890.10929,
      17.38716506,
    },
  },
  {
    T_high = 20000,
    T_low = 6000,
    coeffs = {
      497529430,
      -286610.6874,
      66.9035225,
      -0.00616995902,
      3.016396027e-07,
      -7.4214166e-12,
      7.27817577e-17,
      2293554.027,
      -553.062161,
    },
  },
  ref = "from CEA2::thermo.inp",
}
N2 = {}
```

```
N2.atomic_constituents = {
N=2,
}
N2.charge = 0
N2.M = {
  value = 0.0280134,
  reference = "from CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
N2.d = {
  value = 3.667e-10,
  reference = "Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard sphere diameter, based on L-J parameters",
  units = "m",
}
N2.viscosity = {
  ref = "from CEA2::trans.inp which cites Boushehri et al (1987) and Svehla
(1994)",
  parameters = {
    {
      A = 0.62526577,
      C = -1640.7983,
      B = -31.779652,
      T_high = 1000,
      T_low = 200,
      D = 1.7454992,
    },
    {
      A = 0.87395209,
      C = -173948.09,
      B = 561.52222,
      T_high = 5000,
      T_low = 1000,
      D = -0.39335958,
    },
    {
      A = 0.88503551,
      C = -731290.61,
      B = 909.02171,
      T_high = 15000,
      T_low = 5000,
      D = -0.53503838,
    },
  },
  model = "CEA",
}
N2.thermal_conductivity = {
  ref = "from CEA2::trans.inp which cites Boushehri et al (1987) and Svehla
(1994)",
  parameters = {
    {
      A = 0.85439436,
      C = -12347.848,
      B = 105.73224,
      T_high = 1000,
      T_low = 200,
      D = 0.47793128,
    },
    {
      A = 0.88407146,
      C = -11429.64,
      B = 133.57293,
```

```
      T_high = 5000,
      T_low = 1000,
      D = 0.24417019,
    },
    {
      A = 2.4176185,
      C = 3105580.2,
      B = 8047.7749,
      T_high = 15000,
      T_low = 5000,
      D = -14.517761,
    },
  },
  model = "CEA",
}
N2.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      22103.71497,
      -381.846182,
      6.08273836,
      -0.00853091441,
      1.384646189e-05,
      -9.62579362e-09,
      2.519705809e-12,
      710.846086,
      -10.76003744,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      587712.406,
      -2239.249073,
      6.06694922,
      -0.00061396855,
      1.491806679e-07,
      -1.923105485e-11,
      1.061954386e-15,
      12832.10415,
      -15.86640027,
    },
  },
  {
    T_high = 20000,
    T_low = 6000,
    coeffs = {
      831013916,
      -642073.354,
      202.0264635,
      -0.03065092046,
      2.486903333e-06,
      -9.70595411e-11,
      1.437538881e-15,
      4938707.04,
      -1672.09974,
    },
  },
  ref = "from CEA2::thermo.inp",
}
```

```
H = {}
H.atomic_constituents = {
H=1,
}
H.charge = 0
H.M = {
  value = 0.00100794,
  reference = "CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
H.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
  units = "m",
}
H.viscosity = {
  parameters = {
    {
      A = 0.74226149,
      C = 185541.65,
      B = -401.32865,
      T_high = 5000,
      T_low = 1000,
      D = 0.046741844,
    },
    {
      A = 0.87486623,
      C = 7095504.8,
      B = -2502.2902,
      T_high = 15000,
      T_low = 5000,
      D = -0.93888455,
    },
    ref = "from CEA2::trans.inp which cites Vanderslice et al. (1962)",
  },
  model = "CEA",
}
H.thermal_conductivity = {
  parameters = {
    {
      A = 0.74166119,
      C = 187756.42,
      B = -404.87203,
      T_high = 5000,
      T_low = 1000,
      D = 3.4843121,
    },
    {
      A = 0.87447639,
      C = 7108129.4,
      B = -2508.9452,
      T_high = 15000,
      T_low = 5000,
      D = 2.4970991,
    },
    ref = "from CEA2::trans.inp which cites Vanderslice et al. (1962)",
  },
  model = "CEA",
}
H.CEA_coeffs = {
```

```
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      0,
      0,
      2.5,
      0,
      0,
      0,
      0,
      25473.70801,
      -0.446682853,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      60.7877425,
      -0.1819354417,
      2.500211817,
      -1.226512864e-07,
      3.73287633e-11,
      -5.68774456e-15,
      3.410210197e-19,
      25474.86398,
      -0.448191777,
    },
  },
  {
    T_high = 20000,
    T_low = 6000,
    coeffs = {
      217375769.4,
      -131203.5403,
      33.991742,
      -0.00381399968,
      2.432854837e-07,
      -7.69427554e-12,
      9.64410563e-17,
      1067638.086,
      -274.2301051,
    },
  },
  ref = "from CEA2::thermo.inp",
}
H2 = {}
H2.atomic_constituents = {
H=2,
}
H2.charge = 0
H2.M = {
  value = 0.00201588,
  reference = "molecular weight from CEA2",
  description = "molecular mass",
  units = "kg/mol",
}
H2.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
```

```
  units = "m",
}
H2.viscosity = {
  ref = "from CEA2::trans.inp which cites ASSAEL ET AL (1986)  SVEHLA
(1994)",
  parameters = {
    {
      A = 0.74553182,
      C = -3257.934,
      B = 43.555109,
      T_high = 1000,
      T_low = 200,
      D = 0.13556243,
    },
    {
      A = 0.96730605,
      C = -210251.79,
      B = 679.31897,
      T_high = 5000,
      T_low = 1000,
      D = -1.8251697,
    },
    {
      A = 1.0126129,
      C = -1442848.4,
      B = 1497.3739,
      T_high = 15000,
      T_low = 5000,
      D = -2.3254928,
    },
  },
  model = "CEA",
}
H2.thermal_conductivity = {
  ref = "from CEA2::trans.inp which cites ASSAEL ET AL (1986)  SVEHLA
(1994)",
  parameters = {
    {
      A = 1.0059461,
      C = -29792.018,
      B = 279.51262,
      T_high = 1000,
      T_low = 200,
      D = 1.1996252,
    },
    {
      A = 1.058245,
      C = 11736.907,
      B = 248.75372,
      T_high = 5000,
      T_low = 1000,
      D = 0.82758695,
    },
    {
      A = -0.2236442,
      C = -77771.313,
      B = -6965.0442,
      T_high = 15000,
      T_low = 5000,
      D = 13.189369,
    },
  },
  model = "CEA",
```

```
}
H2.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      40783.2321,
      -800.918604,
      8.21470201,
      -0.01269714457,
      1.753605076e-05,
      -1.20286027e-08,
      3.36809349e-12,
      2682.484665,
      -30.43788844,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      560812.801,
      -837.150474,
      2.975364532,
      0.001252249124,
      -3.74071619e-07,
      5.9366252e-11,
      -3.6069941e-15,
      5339.82441,
      -2.202774769,
    },
  },
  {
    T_high = 20000,
    T_low = 6000,
    coeffs = {
      496688412,
      -314754.7149,
      79.8412188,
      -0.00841478921,
      4.75324835e-07,
      -1.371873492e-11,
      1.605461756e-16,
      2488433.516,
      -669.572811,
    },
  },
  ref = "from CEA2::thermo.inp",
}
H2O = {}
H2O.atomic_constituents = {
O=1, H=2,
}
H2O.charge = 0
H2O.M = {
  value = 0.01801528,
  reference = "CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
H2O.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
```

```
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
  units = "m",
}
H2O.viscosity = {
  parameters = {
    {
      A = 0.50019557,
      C = 88163.892,
      B = -697.12796,
      T_high = 1073.2,
      T_low = 373.2,
      D = 3.0836508,
    },
    {
      A = 0.58988538,
      C = 54263.513,
      B = -537.69814,
      T_high = 5000,
      T_low = 1073.2,
      D = 2.3386375,
    },
    {
      A = 0.64330087,
      C = -377422.83,
      B = -95.668913,
      T_high = 15000,
      T_low = 5000,
      D = 1.812519,
    },
    ref = "from CEA2::trans.inp which cites Sengers and Watson (1986)",
  },
  model = "CEA",
}
H2O.thermal_conductivity = {
  parameters = {
    {
      A = 1.0966389,
      C = 106234.08,
      B = -555.13429,
      T_high = 1073.2,
      T_low = 373.2,
      D = -0.2466455,
    },
    {
      A = 0.39367933,
      C = 612174.58,
      B = -2252.4226,
      T_high = 5000,
      T_low = 1073.2,
      D = 5.8011317,
    },
    {
      A = -0.41858737,
      C = 19179190,
      B = -14096.649,
      T_high = 15000,
      T_low = 5000,
      D = 14.345613,
    },
    ref = "from CEA2::trans.inp which cites Sengers and Watson (1986)",
  },
  model = "CEA",
```

```
}
H2O.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      -39479.6083,
      575.573102,
      0.931782653,
      0.00722271286,
      -7.34255737e-06,
      4.95504349e-09,
      -1.336933246e-12,
      -33039.7431,
      17.24205775,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      1034972.096,
      -2412.698562,
      4.64611078,
      0.002291998307,
      -6.83683048e-07,
      9.42646893e-11,
      -4.82238053e-15,
      -13842.86509,
      -7.97814851,
    },
  },
  ref = "from CEA2::thermo.inp",
}
HO2 = {}
HO2.atomic_constituents = {
O=2, H=1,
}
HO2.charge = 0
HO2.M = {
  value = 0.03300674,
  reference = "CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
HO2.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
  units = "m",
}
HO2.viscosity = {
  parameters = {
    T_ref = 273,
    ref = "Table 1-2, White (2006)",
    S = 111,
    mu_ref = 1.716e-05,
  },
  model = "Sutherland",
}
HO2.thermal_conductivity = {
  parameters = {
```

```
    S = 194,
    ref = "Table 1-3, White (2006)",
    k_ref = 0.0241,
    T_ref = 273,
  },
  model = "Sutherland",
}
HO2.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      -75988.8254,
      1329.383918,
      -4.67738824,
      0.02508308202,
      -3.006551588e-05,
      1.895600056e-08,
      -4.82856739e-12,
      -5873.35096,
      51.9360214,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      -1810669.724,
      4963.19203,
      -1.039498992,
      0.00456014853,
      -1.061859447e-06,
      1.144567878e-10,
      -4.76306416e-15,
      -32008.1719,
      40.6685092,
    },
  },
  ref = "from CEA2::thermo.inp",
}
OH = {}
OH.atomic_constituents = {
O=1, H=1,
}
OH.charge = 0
OH.M = {
  value = 0.01700734,
  reference = "CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
OH.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
  units = "m",
}
OH.viscosity = {
  parameters = {
    {
      A = 0.59711536,
      C = 37606.286,
```

```
      B = -461.00678,
      T_high = 5000,
      T_low = 1000,
      D = 2.4041761,
    },
    {
      A = 0.64287721,
      C = -88543.767,
      B = -181.73747,
      T_high = 15000,
      T_low = 5000,
      D = 1.9636057,
    },
    ref = "from CEA2::trans.inp which cites Svehla (1994)",
  },
  model = "CEA",
}
OH.thermal_conductivity = {
  parameters = {
    {
      A = 0.68627561,
      C = 27559.033,
      B = -740.33274,
      T_high = 5000,
      T_low = 1000,
      D = 2.8308741,
    },
    {
      A = -0.47918112,
      C = 7050995.2,
      B = -9376.9908,
      T_high = 15000,
      T_low = 5000,
      D = 14.203688,
    },
    ref = "from CEA2::trans.inp which cites Svehla (1994)",
  },
  model = "CEA",
}
OH.CEA_coeffs = {
  {
    T_high = 1000,
    T_low = 200,
    coeffs = {
      -1998.85899,
      93.0013616,
      3.050854229,
      0.001529529288,
      -3.157890998e-06,
      3.31544618e-09,
      -1.138762683e-12,
      2991.214235,
      4.67411079,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      1017393.379,
      -2509.957276,
      5.11654786,
      0.000130529993,
```

```
      -8.28432226e-08,
      2.006475941e-11,
      -1.556993656e-15,
      20196.40206,
      -11.01282337,
    },
  },
  {
    T_high = 20000,
    T_low = 6000,
    coeffs = {
      284723419.3,
      -185953.2612,
      50.082409,
      -0.00514237498,
      2.875536589e-07,
      -8.22881796e-12,
      9.56722902e-17,
      1468393.908,
      -402.355558,
    },
  },
  ref = "from CEA2::thermo.inp",
}
H2O2 = {}
H2O2.atomic_constituents = {
O=2, H=2,
}
H2O2.charge = 0
H2O2.M = {
  value = 0.03401468,
  reference = "CEA2::thermo.inp",
  description = "molecular mass",
  units = "kg/mol",
}
H2O2.d = {
  value = 3.617e-10,
  reference = "value for air: Bird, Stewart and Lightfoot (2001), p. 864",
  description = "equivalent hard-sphere diameter, sigma from L-J
parameters",
  units = "m",
}
H2O2.viscosity = {
  parameters = {
    T_ref = 273,
    ref = "Table 1-2, White (2006)",
    S = 111,
    mu_ref = 1.716e-05,
  },
  model = "Sutherland",
}
H2O2.thermal_conductivity = {
  parameters = {
    S = 194,
    ref = "Table 1-3, White (2006)",
    k_ref = 0.0241,
    T_ref = 273,
  },
  model = "Sutherland",
}
H2O2.CEA_coeffs = {
  {
    T_high = 1000,
```

```
    T_low = 200,
    coeffs = {
      -92795.3358,
      1564.748385,
      -5.97646014,
      0.0327074452,
      -3.93219326e-05,
      2.509255235e-08,
      -6.46504529e-12,
      -24940.04728,
      58.7717418,
    },
  },
  {
    T_high = 6000,
    T_low = 1000,
    coeffs = {
      1489428.027,
      -5170.82178,
      11.2820497,
      -8.04239779e-05,
      -1.818383769e-08,
      6.94726559e-12,
      -4.8278319e-16,
      14182.51038,
      -46.5085566,
    },
  },
  ref = "from CEA2::thermo.inp",
}
```

Appendix G: Chemical Kinetics Code

```
-- Bittker_Scullin.py
--
-- This file provides a chemical kinetic description
-- of hydrogen combustion in air.
--
-- The numbering of reactions in this file corresponds
-- to the scheme on p.85 of Bittker and Scullin
--
-- Reference:
--  D. A. Bittker and V. J. Scullin
--  General Chemical Kinetics Computer Program for Static
--  and Flow Reactions, with Application to Combustion
--  and Shock Tube Kinetics
--  NASA TN D-6586 1972
--
-- This file prepared by..
-- Fabian Zander
-- 28th Oct 2010
--
-- a 9-species, 18-reaction scheme
-- Species included: H, H2, O, O2, N2, OH, H2O, HO2, H2O2

reaction{
   'H2 + OH <=> H2O + H',
   fr={'Arrhenius', A=2.10e13, n=0.0, T_a=5100.0/1.987},
   label='r1'
}

reaction{
   'H + O2 <=> OH + O',
   fr={'Arrhenius', A=1.25e14, n=0.0, T_a=16300.0/1.987},
   label='r2'
}

reaction{
   'O + H2 <=> OH + H',
   fr={'Arrhenius', A=2.95e13, n=1.0, T_a=9800.0/1.987},
   label='r3'
}

reaction{
   'H + O2 + M <=> HO2 + M',
   fr={'Arrhenius', A=1.59e15, n=0.00, T_a=-1000.0/1.987},
   label='r4',
   efficiencies={['H2']=5.0, ['H2O']=32.5, ['O2']=2.0, ['N2']=2.0}
}

reaction{
   'H + H + M <=> H2 + M',
   fr={'Arrhenius', A=1.0e18, n=-1.0, T_a=0.0/1.987},
   label='r5',
   efficiencies={['H2']=5.0,['H2O']=15.0, ['O2']=2.0, ['N2']=2.0}
}

reaction{
   'H2 + HO2 <=> H2O2 + H',
   fr={'Arrhenius', A=9.6e12, n=0.00, T_a=24000.0/1.987},
   label='r6',
}
```

```
reaction{
   'M + H2O2 <=> OH + OH + M',
   fr={'Arrhenius', A=1.17e17, n=0.00, T_a=45500.0/1.987},
   efficiencies={['H2O2']=6.6, ['H2']=2.3, ['H2O']=6.0, ['O2']=0.78},
   label='r7',
}

reaction{
   'HO2 + H <=> OH + OH',
   fr={'Arrhenius', A=7.0e13, n=0.00, T_a=0.0/1.987},
   label='r8',
}

reaction{
   'H + OH + M <=> H2O + M',
   fr={'Arrhenius', A=7.50e23, n=-2.6, T_a=0.0/1.987},
   label='r9',
   efficiencies={['H2']=4.0, ['H2O']=20.0, ['O2']=1.6, ['N2']=1.6}
}

reaction{
   'O + O + M <=> O2 + M',
   fr={'Arrhenius', A=1.38e18, n=-1.00, T_a=340.0/1.987},
   label='r10',
}

reaction{
   'O + H2O <=> OH + OH',
   fr={'Arrhenius', A=5.75e13, n=0.00, T_a=18000.0/1.987},
   label='r11',
}

reaction{
   'H2 + O2 <=> OH + OH',
   fr={'Arrhenius', A=1.00e13, n=0.0, T_a=43000.0/1.987},
   label='r12'
}

reaction{
   'HO2 + OH <=> H2O + O2',
   fr={'Arrhenius', A=6.30e12, n=0.00, T_a=0.0/1.987},
   label='r13',
}

reaction{
   'HO2 + O <=> O2 + OH',
   fr={'Arrhenius', A=6.00e12, n=0.00, T_a=0.0/1.987},
   label='r14',
}

reaction{
   'HO2 + HO2 <=> H2O2 + O2',
   fr={'Arrhenius', A=1.80e12, n=0.00, T_a=0.0/1.987},
   label='r15',
}

reaction{
   'OH + H2O2 <=> H2O + HO2',
   fr={'Arrhenius', A=1.00e13, n=0.00, T_a=1800.0/1.987},
   label='r16',
```

```
}

reaction{
   'O + H2O2 <=> OH + HO2',
   fr={'Arrhenius', A=8.00e13, n=0.00, T_a=1000.0/1.987},
   label='r17',
}

reaction{
   'H + H2O2 <=> H2O + OH',
   fr={'Arrhenius', A=3.18e14, n=0.00, T_a=9000.0/1.987},
   label='r18',
}
```